

# SDP Protocol Module for TTCN-3 Toolset with TITAN, Description

Gábor Szalai

Version 1551- CNL 113 353, Rev. C, 2016-08-10

# Table of Contents

How to Read This Document .....	1
Presumed Knowledge .....	1
System Requirements .....	1
Functionality .....	1
Protocol Version Implemented .....	1
Routing Functionality .....	2
Modified and Non-Implemented Protocol Elements .....	2
Relaxed Conditions .....	2
Ericsson-Specific Changes .....	2
Backward Incompatibilities .....	2
Protocol Modifications/Deviations .....	2
Feature List .....	3
Encoding/Decoding and Other Related Functions .....	3
Protocol Modules .....	4
Overview .....	4
Installation .....	4
Configuration .....	4
Parser Generation Rules .....	5
Terminology .....	5
Abbreviations .....	5
References .....	5

# How to Read This Document

This is the User Guide for the SDP protocol module. The SDP protocol module is developed for the TTCN-3 Toolset with TITAN.

## Presumed Knowledge

To use this protocol module the knowledge of the TTCN-3 language [2] is essential.

## System Requirements

Protocol modules are a set of TTCN-3 source code files that can be used as part of TTCN-3 test suites only. Hence, protocol modules alone do not put specific requirements on the system used. However in order to compile and execute a TTCN-3 test suite using the set of protocol modules the following system requirements must be satisfied:

- TITAN TTCN-3 Test Executor R7A (1.7.pl0) or higher installed. For installation guide see [3].

### NOTE

This version of the protocol module is not compatible with TITAN releases earlier than R7A.

## Functionality

## Protocol Version Implemented

This set of protocol modules implements protocol messages and constants of the SDP protocol, (see [1]) with the modifications specified in 3.2. Besides the SDP attributes listed in [1] the following SDP attributes are implemented:

RFC	Attributes
draft-ietf-simple-message-sessions-12 The Message Session Relay Protocol [5]	accept-types, accept-wrapped-types, max-size, path
RFC 3890 A Transport Independent Bandwidth Modifier for the Session Description Protocol (SDP) [6]	pathmaxprate
RFC 3388 Grouping of Media Lines in the Session Description Protocol (SDP) [7]	mid, group
RFC 4145 TCP-Based Media Transport in the Session Description Protocol (SDP) [8]	setup, connection

RFC	Attributes
RFC 4568 Session Description Protocol (SDP) Security Descriptions for Media Streams	crypto
RFC 5245 Interactive Connectivity Establishment (ICE) <a href="#">[10]</a>	ice-ufrag, ice-pwd, candidate, remote-candidates, ice-lite, ice-mismatch, and ice-options
RFC 4585 Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF) <a href="#">[11]</a>	rtcp-fb
RFC 3362 Real-time Facsimile (T.38) – image/t.38 MIME Sub-type Registration	T38Fax
RFC 5547	file-disposition file-date file-icon file-range
draft-ietf-mmusic-sctp-sdp-08 & draft-ietf-mmusic-sctp-sdp-11	sctpmap, sctp_port, max-message-size

## Routing Functionality

Routing functionality is not performed.

## Modified and Non-Implemented Protocol Elements

### Relaxed Conditions

There is no constraint between received and sent messages.

## Ericsson-Specific Changes

There is no Ericsson specific change in this product.

## Backward Incompatibilities

None.

## Protocol Modifications/Deviations

Protocol modules contain the following modifications/deviations from [\[1\]](#) changing the protocol message structure/behavior:

- In the TITAN executor environment, integer values are stored as 32-bit signed integers. Hence, those variables that represent an integer value greater than  $2^{31}-1$  cannot be stored in a variable of type integer. This affects the following information elements:
  - Time field: start time, stop time
  - Zone adjustment time
  - Origin field: session ID, session version

These integer values are represented as charstrings.

- At most one bandwidth field is allowed at session level and at most one for each media description. (The standard [1] specifies it ambiguously: the grammar allows multiple bandwidth fields, but the textual description does not allow this.)
- The decoder supports all three variants of email-address and phone-number formats (see [1] Appendix A). By default, the encoder encodes these fields in the following form: email (displayable name) and phone (displayable name). If you want to encode these fields in the form: displayable name <email> and displayable name <phone>, use the *f\_SDP\_enc\_Message\_RFC822Name* encoder method for encoding (see [4]).

The language specified by [1] can not be used for creating an LR(1) parser. Hence, this implementation contains a simplified parser, which does not constrain the format of text fields. This should not cause any trouble when decoding SDP messages. However, content of fields should be verified in TTCN, when needed.

The encoder does not perform any format check on charstring fields. It is the user's responsibility to ensure that charstring fields contain valid data.

## Feature List

### Encoding/Decoding and Other Related Functions

This product contains encoding/decoding functions, which assure correct encoding of messages when sent from Titan and correct decoding of messages when received by Titan. Implemented encoding/decoding functions:

Name	Type of formal parameters	Type of return value
<i>f_SDP_enc_Message</i>	(in SDP_Message)	return charstring
<i>f_SDP_enc_Message_RFC822Name</i>	(in SDP_Message)	return charstring
<i>f_SDP_dec_Message</i>	(in charstring)	return SDP_Message
<i>f_SDP_decodeMessage</i>	(in charstring input, out SDP_Message msg)	return integer
<i>f_SDP_enc_SRTP_Crypto</i>	(in SRTP_Crypto input)	return charstring;

Name	Type of formal parameters	Type of return value
<code>f_SDP_dec_SRTP_Crypto</code>	(in charstring input)	return SRTP_Crypto

The function `f_SDP_dec_Message` decodes charstrings that represent an encoded SDP message. The function `f_SDP_enc_Message` encodes an SDP message given as TTCN-3 data into a character string. The function `f_SDP_enc_Message_RFC822Name` performs the same task as the `f_SDP_enc_Message`, but the email/phone address will be in form that RFC 822 specifies: displayable name `<email>` and displayable name `<phone>` (see [5]).

# Protocol Modules

## Overview

Protocol modules implement the message structures of the related protocol in a formalized way, using the standard specification language TTCN-3 . This allows defining of test data (templates) in the TTCN-3 language [6] and correctly encoding/decoding messages when executing test suites using the Titan TTCN-3 test environment [4].

Protocol modules are using Titan's RAW encoding attributes [4] and hence is usable with the Titan test toolset only.

## Installation

The set of protocol modules can be used in developing TTCN-3 test suites using any text editor. However to make the work more efficient a TTCN-3-enabled text editor is recommended (e.g. `nedit`, `xemacs`). Since the SDP protocol is used as a part of a TTCN-3 test suite, this requires TTCN-3 Test Executor be installed before the module can be compiled and executed together with other parts of the test suite. For more details on the installation of TTCN-3 Test Executor see the relevant section of [4].

Release of the protocol module contains a demo directory. The demo directory contains 3 files:

- `SDP.prj` - Titan GUI project file
- `SDP_Templates.ttcn` - Test SDP messages
- `SDP_test.ttcn` - Encoding/decoding test case to encode/decode messages listed in `SDP_Templates.ttcn`.

## Configuration

None.

# Parser Generation Rules

In order to generate the `.c` and `.h` files from `.y` and `.l` the following *Makefile* rules must be used:

```
SDP_parse_.tab.c SDP_parse_.tab.h: SDP_parser.y  
  
bison -dv -p SDP_parse_ -b SDP_parse_ $<  
  
lex.SDP_parse_.c: SDP_parser.l  
  
flex -Cfr -8 -Bvpp -PSDP_parse_ SDP_parser.l
```

The `.h` and `.c` parser files should be generated during the protocol module development. Only the pregenerated files are needed for test case development and test execution.

## Terminology

No specific terminology used.

## Abbreviations

### IP

Internet Protocol

### SDP

Session Description Protocol

### TTCN-3

Testing and Test Control Notation version 3

## References

[1] [RFC 2327](#)

SDP: Session Description Protocol

[2] ETSI ES 201 873-1 v.2.2.1 (02/2003)

The Testing and Test Control Notation version 3. Part 1: Core Language

[3] User Documentation for the TITAN TTCN-3 Test Executor

[4] [RFC 822](#)

Standard for the format of ARPA Internet text messages

[5] <https://tools.ietf.org/html/draft-ietf-simple-message-sessions-12>

The Message Session Relay Protocol

[6] [RFC 3890](#)

A Transport Independent Bandwidth Modifier for the Session Description Protocol (SDP)

[7] [RFC3388](#)

Grouping of Media Lines in the Session Description Protocol (SDP)

[8] [RFC 4145](#)

TCP-Based Media Transport in the Session Description Protocol (SDP)

[9] [RFC 4568](#)

Session Description Protocol (SDP) Security Descriptions for Media Streams

[10] [RFC 5245](#)

Interactive Connectivity Establishment (ICE)

[11] [RFC 4585](#)

Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)

[12] [RFC 3362](#)

Real-time Facsimile (T.38) – image/t.38 MIME Sub-type Registration