JSON v07 2006 Protocol Modules for TTCN-3 Toolset with Titan, Description

Gábor Szalai

Table of Contents

Functionality

The JSON v07 2006 protocol module implements the message structures of the related protocol [4] in a formalized way, using the standard specification language TTCN-3. This allows defining of test data (templates) in the TTCN-3 language and correctly encoding/decoding messages when executing test suites using the Titan TTCN-3 test environment.

The JSON v07 2006 protocol module uses Titan's TEXT encoding attributes [3] and hence is usable with the Titan test toolset only.

Implemented Protocols

This set of protocol modules implements protocol messages and constants of the JSON protocol as described in [4].

Modified and Non-Implemented Protocol Elements

Numbers:
Numbers can only be integers with the digits specified. ".", "e+", "e-", "E+", "E-" are not handled. Fractions and exponentials are discarded when decoding.

Ericsson-Specific Changes

None.

Backward Incompatibilities

The product is not backward compatible with CNL 113 676 because of the following changes:

- The encoder/decoder function takes the encoded form as octetstring instead of charstring
- The `JSON_String` is represented as universal charstring instead of charstring

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 version CRL 113 200/5 R3A (5.3.pl0) or higher installed. For Installation Guide see [2].

| NOTE | This version of the test port is not compatible with Titan releases earlier than CRL 113 200 R7A. |

Usage

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 TTCN3enabled text editor is recommended (for example, nedit, xemacs). Since the JSON 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 [2].

Configuration

None.

Examples

None.

Interface Description

Top Level PDU

The top level PDUs is the TTCN-3 union JSON_PDU.

Encoding/Decoding and Other Related Functions

This product also 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 and Decoding Functions

Name	Type of formal parameters	Type of return value
enc_JSON	JSON_PDU pdu	octetstring
dec_JSON	octetstring msg	JSON_PDU
dec_JSON_Sliding	octetstring msg, JSON_PDU pdu	integer
enc_JSON_escaped	JSON_PDU pdu	octetstring
dec_JSON_escaped	octetstring msg	JSON_PDU

Name	Type of formal parameters	Type of return value
dec_JSON_Sliding_escaped	octetstring msg, JSON_PDU pdu	integer

The usage of the *escaped* functions is recommended, since they automatically handle escaped characters in strings.

Terminology

Abbreviations

PDU

Protocol Data Unit

JSON

JavaScript Object Notation

TTCN-3

Testing and Test Control Notation version 3

References

[1] ETSI ES 201 873-1 v4.5.1 (2013-04)
The Testing and Test Control Notation version 3. Part 1: Core Language

[2] User Guide for TITAN TTCN-3 Test Executor

[3] Programmer's Technical Reference for Titan TTCN–3 Test Executor

[4] IETF RFC 4627
The application/json Media Type for JavaScript Object Notation (JSON)July 2006