

WebSocket protocol module for TTCN-3 Toolset with TITAN, Function Description

Gábor Szalai

Version 1551-CNL 113 782, Rev. A, 2013-05-17

Table of Contents

Functionality	1
Implemented Protocols	1
System Requirements	1
Installation	1
Encoder/Decoder Functions	1
Masking of the Payload Data	1
Masking Key Generation	2
Message Length Calculation Function	2
References	2

Functionality

The WebSocket protocol module provides type definitions, encoder, and decoder functions to handle WebSocket messages defined by the standard [\[2\]](#).

Implemented Protocols

The WebSocket protocol module implements all the messages, and information elements defined in [RFC 6455](#).

System Requirements

In order to operate the WebSocket test port the following system requirements must be satisfied:

- TITAN TTCN-3 Test Executor version R8B (1.8.pl1) or higher installed.

NOTE | This version of the test port is not compatible with TITAN releases earlier than R8B.

Installation

Since the WebSocket test port is used as a part of the TTCN-3 test environment this requires TTCN-3 Test Executor to be installed before any operation of the WebSocket test port. For more details on the installation of TTCN-3 Test Executor see the relevant section of [\[3\]](#).

Encoder/Decoder Functions

The WebSocket protocol module declares the following encoder, and decoder functions:

```
external function f_WebSocket_Encode(in WebSocket_PDU pl_pdu, out octetstring
pl_data,in boolean pl_gen_maks:=m_Websocket_generate_masking_key, in boolean
pl_auto_maks:= m_Websocket_auto_masking);

external function f_WebSocket_Decode(in octetstring pl_data, out WebSocket_PDU pl_pdu,
in boolean pl_auto_maks:= m_Websocket_auto_masking) return integer;
```

Masking of the Payload Data

The encoder and decoder functions are able to mask or unmask the Payload data of the WebSocket messages. The functionality is controlled by the `pl_auto_maks` parameter. If the `pl_auto_maks` is true and masking key is present in the message the payload data is masked or unmasked.

Masking Key Generation

The masking key can be generated by the following function or auto generated by the `f_WebSocket_Encode` function during encoding.

```
external function f_WebSocket_Generate_Masking_Key() return octetstring;
```

The function returns a randomly generated 4 octet length octetstring.

The `f_WebSocket_Encode` function automatically generates and inserts the masking key into the message if:

1. The `pl_gen_maks` parameter is `true`
2. And `mask_bit == '1'B`
3. And `masking_key` is `omit` or `masking_key=='00000000'0`

Message Length Calculation Function

The following function can be used to calculate the length of the received message. The function returns the length of the received message in octets or `-1` if the length can't be calculated.

```
external function f_WebSocket_calc_length(in octetstring pl_data) return integer;
```

References

[1] ETSI ES 201 873-1 v4.5.1

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

[2] [RFC 6455](#)

The WebSocket Protocol

[3] Programmer's Technical Reference for TITAN TTCN-3 Test Executor