

# ICMPv6 Protocol Modules for TTCN-3 Toolset with TITAN, User Guide

János Kövesdi

Version 198 17-CNL 113 631, Rev. A, 2013-09-06

# Table of Contents

About This Document .....	1
How to Read This Document .....	1
Presumed Knowledge .....	1
System Requirements .....	1
Protocol Modules .....	1
Overview .....	1
Installation .....	2
Configuration .....	2
Encoding, Decoding, Checksum Calculation .....	2
Terminology .....	3
Abbreviations .....	3
References .....	3

# About This Document

## How to Read This Document

This is the User Guide for the ICMPv6 protocol module. The ICMPv6 protocol module is developed for the TTCN-3 Toolset with TITAN. This document should be read together with Function Specification [4].

## Presumed Knowledge

To use this protocol module the knowledge of the TTCN-3 language [1] 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 1.8.pl0.pre2 or higher installed. For installation guide see [3].

### NOTE

This version of the protocol module can not be used for defining 32bit unsigned or larger integers with TITAN versions earlier than 1.8.pl0.pre2.  
This version of the protocol module is not compatible with TITAN releases earlier than R7A.

# Protocol Modules

## Overview

Protocol modules implement the message structures of the corresponding protocol in a formalized way, using the standard specification language TTCN-3. This allows defining of test data (templates) in the TTCN-3 language [1] and correctly encoding/decoding messages when executing test suites using the TITAN TTCN-3 test environment.

Protocol modules are using TITAN's RAW encoding attributes [2] and hence are usable with the TITAN test toolset only.

The table below contains the implemented ICMPv6 messages and the corresponding TTCN-3 type records. Using those type records, templates can be defined to send and receive a given message.

Message name	Reference	Corresponding type record in <i>ICMPv6_Types.ttcn</i>
Destination Unreachable	[5] 3.1	<i>ICMPv6_DestinationUnreachable</i>

Message name	Reference	Corresponding type record in <i>ICMPv6_Types.ttcn</i>
Packet Too Big	[5] 3.2	<i>ICMPv6_PacketTooBig</i>
Time Exceeded Message	[5] 3.3	<i>ICMPv6_TimeExceeded</i>
Parameter Problem	[5] 3.4	<i>ICMPv6_ParameterProblem</i>
Echo Request	[5] 4.1	<i>ICMPv6_EchoRequest</i>
Echo Reply	[5] 4.2	<i>ICMPv6_EchoReply</i>

## 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, for example `nedit`, `xemacs`. Since the ICMPv6 protocol module is used as a part of a TTCN-3 test suite, this requires TTCN-3 Test Executor and a C compiler 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 parts of [2].

## Configuration

None.

## Encoding, Decoding, Checksum Calculation

Implemented encoding/decoding functions:

Name	Type of formal parameters	Type of return value
<i>f_enc_PDU_ICMPv6</i>	PDU_ICMPv6, OCT16, OCT16	octetstring
<i>f_dec_PDU_ICMPv6</i>	octetstring	PDU_ICMPv6

The encoding function *f\_enc\_PDU\_ICMPv6* performs basic RAW encoding [2]. In addition to encoding functionality this function can calculate the checksum field. The checksum is automatically calculated during encoding if the user sets the *PDU\_ICMPv6's checksum* field to `'0000'0`. For calculating the correct checksum the user needs to provide 2 additional parameters to the encoding function. These parameters are the *srcaddr* and *dstaddr*. The value of *srcaddr* is the source address in the IPv6 packet. The value of *dstaddr* is the destination address in the IPv6 packet. The default value for *srcaddr* is the dummy value `'00000000000000000000000000000000'0`. The default value for *dstaddr* is the dummy value `'00000000000000000000000000000000'0`. If the user sets the *PDU\_ICMPv6's checksum* field to a value different from `'0000'0` then this user defined value will appear in the encoded message.

The decoding function *f\_dec\_PDU\_ICMPv6* performs basic RAW decoding [2]. The *checksum* field is not verified during decoding.

There is an additional function which verifies the *checksum* field in an encoded ICMPv6 message:

Name	Type of formal parameters	Type of return value
<code>f_ICMPv6_verify_checksum</code>	octetstring, OCT16, OCT16	boolean

The inputs into this function are the encoded ICMPv6 message and the source and destination addresses of the IPv6 packet. The default value for `srcaddr` is the dummy value `'00000000000000000000000000000000'`. The default value for `dstaddr` is the dummy value `'00000000000000000000000000000000'`. The function returns `true` if the checksum is correct and `false` if it is incorrect.

# Terminology

TITAN TTCN-3 Test Executor.

# Abbreviations

## IETF

Internet Engineering Task Force

## IP

Internet Protocol

## ICMPv6

Internet Control Message Protocol for IPv6

## IPv6

Internet Protocol Version 6

## RFC

Request for Comments

## TTCN-3

Testing and Test Control Notation version 3

# References

- [1] ETSI ES 201 873–1 v.3.1.1 (06/2005)  
The Testing and Test Control Notation version 3. Part 1: Core Language
- [2] Programmer’s Technical Reference for the TITAN TTCN-3 Test Executor
- [3] Installation Guide for the TITAN TTCN-3 Test Executor
- [4] ICMPv6 Protocol Modules for TTCN-3 Toolset with TITAN, Function Specification
- [5] IETF [RFC 4443](#)

