|  |  |
| --- | --- |
|  |  |

SNMP ASN.1 protocol module for TTCN-3 Toolset with TITAN, Function Description

# Abstract

The purpose of this document is to specify the functionality and usage of the SNMP ASN.1 protocol module.

Contents

[1 Functionality 2](#_Toc352335242)

[1.1 Protocol version implemented 2](#_Toc352335243)

[1.2 Modifications/deviations related to the protocol specification 2](#_Toc352335244)

[1.2.1 Unimplemented Messages, Information Elements and Constants 2](#_Toc352335245)

[1.2.2 Protocol Modifications/Deviations 2](#_Toc352335246)

[1.3 System Requirements 2](#_Toc352335247)

[2 Usage 3](#_Toc352335248)

[2.1 Installation 3](#_Toc352335249)

[2.2 Examples 3](#_Toc352335250)

[3 Interface Description 3](#_Toc352335251)

[3.1 Top Level PDU 3](#_Toc352335252)

[3.2 Encoder, decoder functions 3](#_Toc352335253)

[3.2.1 Implemented encoding/decoding functions: 4](#_Toc352335254)

[4 Terminology 6](#_Toc352335255)

[5 References 6](#_Toc352335256)

[6 Change information 7](#_Toc352335257)

# Introduction

This is the Function Description for the set of SNMP ASN.1 protocol modules. The SNMP ASN.1 protocol modules are developed for the TTCN-3 Toolset with TITAN. This document should be read together with Product Revision Information [2].

# Functionality

Protocol modules implement the message structures of the related protocol in a formalized way, using the standard specification language ASN.1 [8]. 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 version implemented

This set of protocol modules implements protocol messages and constants of the SNMP protocol, (see[3][4][5][6]). SNMPv1, SNMPv2 and SNMPv3 messages are supported.

## Modifications/deviations related to the protocol specification

None.

### Unimplemented Messages, Information Elements and Constants

None.

### Protocol Modifications/Deviations

None.

## System Requirements

Protocol modules are a set of ASN.1 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 R7A (1.7.pl0) or higher installed. Please note: This version of the protocol module is not compatible with TITAN releases earlier than 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 TTCN-3-enabled text editor is recommended (e.g. nedit, xemacs). Since the SNMP 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 [7].

## Examples

The demo directory contains the file SNMP\_IPL4\_Definitions.ttcn which defines a dual face port. It can be used with the IPL4 test port [10] to send and receive SNMPv1\_Message, SNMPv2\_Message or SNMPv3\_Message in the test case in the same way it was implemented in the SNMP test port [11]. The only difference is the dynamic update of the remote address. This can be performed now with the IPL4 test port’s f\_IPL4\_Connect function.

The demo directory also contains the file SNMP\_Test.ttcn. This file includes two test cases. The first test case is called tc\_SNMP\_1 which sends and receives an SNMPv1\_Message, an SNMPv2\_Message and an SNMPv3\_Message using the dual face port. The second test case is called tc\_SNMP\_2. It performs the same actions as the first test case, then the remote address is changed, then the actions of the first test case are repeated using the new remote address.

# Interface Description

## Top Level PDU

The top level PDUs are SNMPv1\_Message, SNMPv2\_Message and SNMPv3\_Message.

## Encoder, decoder functions

This product also contains encoding/decoding functions, which assure correct BER [9] encoding of messages when sent from Titan and correct BER [9] decoding of messages when received by Titan.

### Implemented encoding/decoding functions:

The following functions can be used for encoding and decoding of SNMP v1, v2 or v3 messages :

|  |  |  |
| --- | --- | --- |
| **Name** | **Type of formal parameters** | **Type of return value** |
| enc\_SNMPv1\_Message | in SNMPv1\_Message | octetstring |
| dec\_SNMPv1\_Message | in octetstring | SNMPv1\_Message |
| dec\_SNMPv1\_Message\_backtrack | in octetstring, out SNMPv1\_Message | integer (0: success, 1: decoding failed) |
| enc\_SNMPv2\_Message | in SNMPv2\_Message | octetstring |
| dec\_SNMPv2\_Message | in octetstring | SNMPv2\_Message |
| dec\_SNMPv2\_Message\_backtrack | in octetstring, out SNMPv2\_Message | integer (0: success, 1: decoding failed) |
| enc\_SNMPv3\_Message | in SNMPv3\_Message | octetstring |
| dec\_SNMPv3\_Message | in octetstring | SNMPv3\_Message |
| dec\_SNMPv3\_Message\_backtrack | in octetstring, out SNMPv3\_Message | integer (0: success, 1: decoding failed) |

The following functions can be used for object identifier operations:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type of formal parameters** | **Type of return value** | **Functionality** |
| oi\_concat | in objid, in objid | objid | return concatenation of inputs |
| oi\_size | in objid | integer | return number of elements of input |
| oi\_eq | in objid, in objid | boolean | return true if two inputs are equal |
| oi\_sub | in integer, in integer , in objid | objid | return sub-objid where first input is begin and second input is size |
| oi\_gt | in objid, in objid | boolean | return true if first input > second input lexicographically |
| oi\_addObjId | in objid, in SetOfInteger | Objid | return concatenation of first input with integers of second input |
| oi\_setvalue | in objid, in integer, in integer | objid | insert into first input at position specified by the second input a value given by the third input |
| oi\_getvalue | in objid input, in integer position | integer | return element selected by second input |

The following functions can be used for authentication and encryption of SNMP v1, v2 or v3 messages :

|  |  |  |
| --- | --- | --- |
| **Name** | **Type of formal parameters** | **Type of return value** |
| calculate\_MD5\_MAC | in octetstring (authentication key), in SNMPv3\_Message (message) | octetstring |
| encryptData | in octetstring (encryption key), in ScopedPDU (data to encrypt, out octetstring (encrypted data),  in octetstring (privacy parameters) | - |
| decryptData | in octetstring (decryption key), in octetstring (privacy parameters), in octetstring (encrypted data),  out ScopedPDU (decrypted data) | - |
| password\_to\_key\_md5 | in octetstring (password), in octetstring (engine ID), out octetstring (key) | - |
| enc\_UsmSecurityParameters | in UsmSecurityParameters | octetstring |
| dec\_UsmSecurityParameters | in octetstring | UsmSecurityParameters |

# Terminology

ASN.1 Abstract Syntax Notation One

ASP Abstract Service Primitive

BER Basic Encoding Rules

SNMP Simple Network Management Protocol

TTCN-3 Testing and Test Control Notation version 3

# References

1. ETSI ES 201 873-1 v4.5.1   
   The Testing and Test Control Notation version 3. Part 1: Core Language
2. 109 21-CNL 113 774-1  
   SNMP ASN.1 Protocol Module for TTCN-3 Toolset with TITAN, Product Revision Information
3. RFC 1157  
   A Simple Network Management Protocol (SNMP)
4. RFC 1901  
   Introduction to Community-based SNMPv2
5. RFC 3411  
   Architecture for SNMP management frameworks
6. RFC 3414  
   User-based Security Model (USM) for version 3 of the Simple Network  
   Management Protocol (SNMPv3)
7. 2/198 17-CRL 113 200/3 Uen  
   Programmer’s Technical Reference for TITAN TTCN–3 Test Executor
8. ITU-T X.680 (07-2002)  
   Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation
9. ITU-T X.690 (07-2002)  
   Information technology – ASN.1 encoding rules:  
   Specification of basic encoding Rules (BER),  
   Canonical encoding rules (CER) and  
   Distinguished encoding rules (DER)
10. 109 21-CNL 113 531-10 Uen  
    IPL4asp Test Port for TTCN-3 Toolset with TITAN, Product Revision Information
11. 109 21-CNL 113 344-4 Uen  
    SNMP Test Port for TTCN – 3 Toolset with TITAN,  
    Revision Information

# Change information

This is the first version of this document.