M3UA Protocol Emulation for TTCN-3 Toolset with TITAN, Function Specification

Ferenc Kovács

Version 198 17-CNL 113 537, Rev. A, 2008-01-14

Table of Contents

About This Document

How to Read This Document

This is the User Guide for the M3UA Protocol Emulation (M3UA PE). The M3UA PE is developed for the TTCN-3 Toolset with TITAN according to the Requirement Specification [5]. This document should be read together with Function Specification [6].

Presumed Knowledge

The knowledge of the TITAN TTCN-3 Test Executor [2] and the TTCN-3 language [1] is essential.

System Requirements

In order to operate the M3UA PE the following system requirements must be satisfied:

- Any kind of Linux distribution with SCTP enabled kernel (our test system was Gentoo (Base System version 1.12.1) with version 2.6.21 SMP kernel).
- LKSCTP [9] library installed (we used version 2.6.16-1.0.6) for the SCTP test port to operate.

The Protocol Emulation

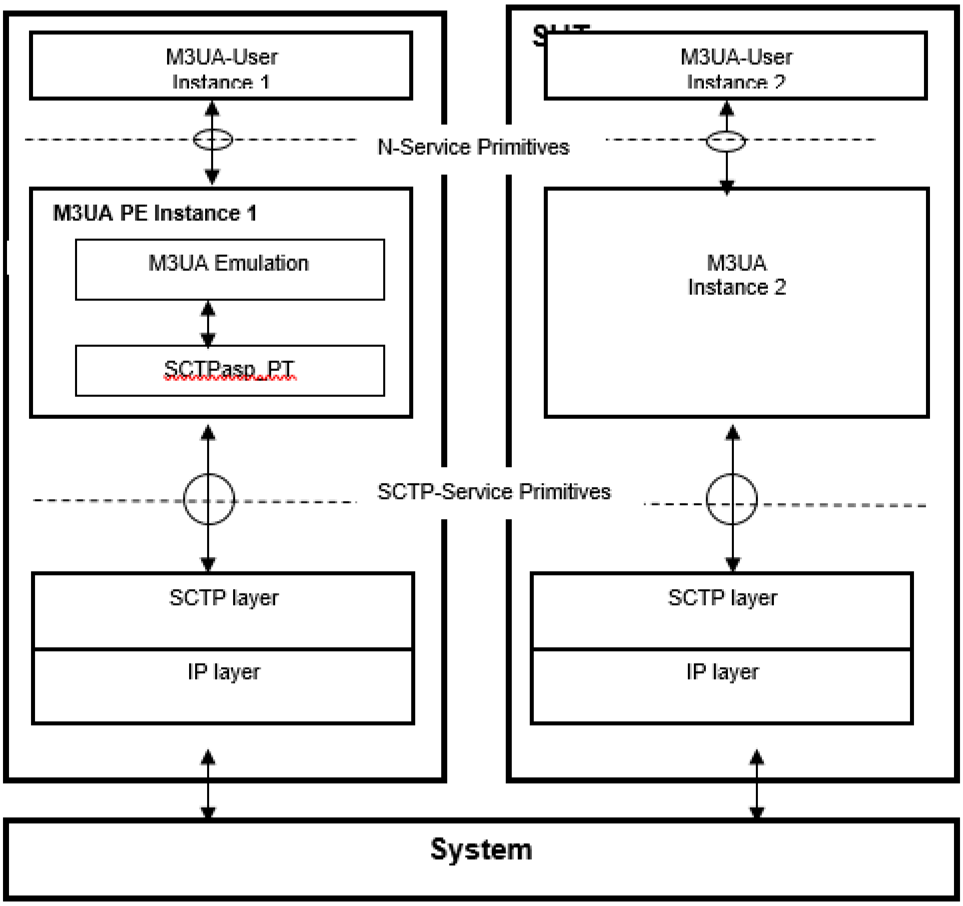
Overview

The M3UA PE is developed for testing implementations of M3UA users using TTCN-3. It is assumed that the M3UA layer of the peer conforms to the same specifications as M3UA PE does.

The M3UA PE is similar to the M3UA SCTP Daemon (see [7]). The most important difference is that, M3UA PE is not a standalone program, it runs until the test case finishes. It handles only one SCTP association instead of many in contrast to the M3UA SCTP Daemon. We store the endpoints (local IP address/port and the remote IP address/port) of this single SCTP association.

The M3UA PE communicates with the M3UA user with MTP3 ASPs. The M3UA PE receives `ASP_MTP3_TRANSFERreq` messages from the M3UA users and transfers them via SCTP to the SUT. The M3UA users receive `ASP_MTP3_TRANSFERind` messages from the emulation, which messages originate from the SUT.

See the service primitives in M3UA PE below:



Installation

Since M3UA PE 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 M3UA PE. For more details on the installation of TTCN-3 Test Executor (see [4]). An implementation of the SCTP test port is also needed for execution.

Description of Files Implementing the M3UA PE

The M3UA PE is implemented in the *M3UA_Emulation.ttcn* file.

To build an executable test suite with TITAN [2] that is using M3UA PE the following files from other products shall be added to the related project.

- *M3UA_Emulation.ttcn* (This product)
- *General_Types.ttcn* (CNL 113 368)
- *MTP3asp_EncDec.cc* (CNL 113 337)
- *MTP3asp_Types.cc* (CNL 113 337)
- *M3UA_Types.ttcn* (CNL 113 536)SCTPasp_PT.cc (CNL 113 469)
- *SCTPasp_PT.hh* (CNL 113 469)
- *SCTPasp_Types.ttcn* (CNL 113 469)
- *SCTPasp_PortType.ttcn* (CNL 113 469)

Their functionality is the following:

- *General_Types.ttcn*
  General type definitions.
- *MTP3asp_EncDec.cc*
  MTP3asp_Types.ttcn contains two external function declarations. The implementation for these functions (`enc_MTP3_to_M3UAserver_msg`() and `dec_MTP3_to_M3UAserver_msg`()) are placed in this file. This is only needed to prevent linking errors. The functions themselves are not used.
- *MTP3asp_Types.ttcn*
  This file contains MTP3 ASP definitions and templates we use (for example, `ASP_MTP3_TRANSFERind`, `ASP_MTP3_TRANSFERreq`, `ASP_MTP3_PAUSE`, `ASP_MTP3_RESUME`, `ASP_MTP3_STATUS`).
- *M3UA_Emulation.ttcn*
  This file contains the dynamical part.
- *M3UA_Types.ttcn*
  M3UA protocol module. This file contains M3UA type definitions for all M3UA message types and the appropriate encoding (RAW) and decoding (RAW) functions.
- `SCTPasp`
  The files of the SCTP test port.

Configuration

The M3UA PE uses module parameters for runtime configuration that can be set in the `[MODULE_PARAMETERS]` section of a TITAN RTE configuration file. There are a few user configurable module parameters, and a few timer settings that are not advised to be altered by the user.

A list of configurable parameters is provided below.

- `tsp_SCTP_Server_Mode` (optional):
  If this parameter is `true` then the M3UA PE doesn’t initiate SCTP associations, but waits for the communication ups from the remote address.
- `tsp_logVerbose` (optional):
  This parameter allows the output of textual debug information of TTCN-3 "log" statements on the console or in log file (depending on the setting of `consoleMask` and `fileMask` parameters).

  NOTE Error messages for serious errors are not affected by the `tsp_logVerbose` parameter.

  The default value of `tsp_logVerbose` is `false`.
- `tsp_Timer` (optional):
  This parameter is used only in client mode. The client will wait for an `ASP_SCTP_RESULT` message from the server for 60 seconds in response of the client’s `ASP_SCTP_ConnectFrom` message.

  Default value is `60` seconds.
- `tsp_Enable_M3UA_Heartbeat` (optional):

This parameter enables the sending of M3UA heartbeats periodically (setting it to `true`).

Default value is `false`.

- `tsp_Heartbeat_Timer` (optional):
  The value of this parameter determines the period for sending M3UA heartbeats, if enabled by `tsp_Enable_M3UA_Heartbeat` parameter.

  Default value is `30` seconds.

- `tsp_ASPUP_Resend_Timer` (optional):
  If we didn't receive any response to our `M3UA_ASPUP` message (i.e. an `M3UA_ASPUP_Ack` message), it must be resent. Used only if `tsp_M3UA_Server_Mode` is `false`.

  Default value is `2` seconds.

- `tsp_ASPAC_Resend_Timer` (optional):
  If we didn't receive any response to our `M3UA_ASPAC` message (i.e. an `M3UA_ASPAC_Ack` message), it must be resent. Used only if `tsp_M3UA_Server_Mode` is `false`.

  Default value is `2` seconds.

- `tsp_Assoc_Restart_Timer` (optional):
  The value of this parameter determines the period for checking if the SCTP connection is lost. If so, then the M3UA PE tries to re-establish the SCTP connection at this period. Used only if `tsp_SCTP_Server_Mode` is `false`.

Starting the Emulation

`f_M3UA_Emulation`() function is the main function of the M3UA PE, which should be called by the user to start M3UA PE, and to set the endpoints of the SCTP association. The parameter of this function is an `SCTP_Association_Address` structure, which looks like the following.

```
type record SCTP_Association_Address
{
  integer local_sctp_port;
  charstring local_ip_address;
  integer remote_sctp_port;
  charstring remote_sctp_address;
}
```

When the user wants to start the emulation in a test case, such a structure must be filled and passed to the `f_M3UA_Emulation`()function. The SCTP test port will build the association using this data.

Logging

The logging mechanism logs information either to the log file or the output console. The amount and type of logging can be set in the RTE configuration file. This setting is on two levels. The TITAN parameters ConsoleMask and FileMask control the overall logging. These logging options are described in Section 7.2 of TITAN's Programmer's Technical Reference (see [3]). The logging options specific to the M3UA PE are controlled by the M3UA PE parameter tsp_logVerbose (see Configuration). If tsp_logVerbose is selected most events are logged textually in addition to the possible logging of the sent/received messages. Since there is an MTC and there can be any number of PTCs within the M3UA PE, there are log files for the MTC itself, and for all PTCs. These files include logging from the M3UA PE and the SCTP test port.

Limitations

- Only a single SCTP association is supported between two endpoints. For example we can connect an IP1 and PORT1 pair to an IP2 and PORT2 pair.
- Sending ASP_MTP3_PAUSE, ASP_MTP3_RESUME, ASP_MTP3_STATUS messages is not supported.
- Multi-homing is not supported.

Examples

Configuration File

This is a simple example configuration file for testing. The relevant information can be found under the sections TESTPORT_PARAMETERS and MODULE_PARAMETERS.

```
[LOGGING]
LogSourceInfo := Yes
FileMask := LOG_ALL | TTCN_DEBUG | TTCN_MATCHING
ConsoleMask := LOG_ALL | TTCN_DEBUG | TTCN_MATCHING

[EXECUTE]
M3UA_Emulation_Test.tc_M3UA_demo_bicc

[TESTPORT_PARAMETERS]
system.SCTP_PORT.debug := "yes"

[MODULE_PARAMETERS]
M3UA_Emulation.tsp_logVerbose := true;
M3UA_Emulation.tsp_M3UA_Server_Mode := false;
M3UA_Emulation_Test.tsp_address :=
{
  local_sctp_port   := 2905,
  local_ip_addr  := "10.6.70.19",
  remote_sctp_port  := 2905,
  remote_ip_addr := "159.107.193.33"
}
```

Test Suite

This is an example how to build up the protocol stack in case of BICC test. The test function on BICC level should be started after `f_testconfig_bicc`() was called.

```
type component BICC_CT
{
  port MTP3asp_PT BICC_MTP3_PORT;
}

type component MTC_CT
{
  var  BICC_CT vlc_BICC_COMPONENT_1;
  var  M3UA_CT vlc_M3UA_COMPONENT_1;
  port SCTPasp_PT SCTP_PORT;
}

function f_testconfig_bicc(SCTP_Association_Address pl_address) runs on MTC_CT
{
  vlc_BICC_COMPONENT_1 := BICC_CT.create;
  vlc_M3UA_COMPONENT_1 := M3UA_CT.create;
  connect(vlc_BICC_COMPONENT_1:BICC_MTP3_PORT,
          vlc_M3UA_COMPONENT_1:MTP3_SP_PORT);
  map(system:SCTP_PORT, vlc_M3UA_COMPONENT_1:SCTP_PORT);
  vlc_M3UA_COMPONENT_1.start(f_M3UA_Emulation(pl_address));
}

function f_testconfig_end_bicc() runs on MTC_CT
{
  unmap(system:SCTP_PORT,
        vlc_M3UA_COMPONENT_1:SCTP_PORT);
  disconnect(vlc_BICC_COMPONENT_1:BICC_MTP3_PORT,
             vlc_M3UA_COMPONENT_1:MTP3_SP_PORT);
  vlc_M3UA_COMPONENT_1.stop;
}

testcase tc_M3UA_demo_bicc() runs on MTC_CT
{
  f_testconfig_bicc(tsp_address);
  vlc_BICC_COMPONENT_1.start(f_BICC());
  vlc_BICC_COMPONENT_1.done;
  f_testconfig_end_bicc();
}
```

Terminology

- **M3UA Protocol Emulation:**

Implementation of M3UA as specified in [6].

- **M3UA User: Protocol:**
  Uses services of M3UA.
- **SCTP Client Mode:**
  The M3UA PE establishes the SCTP associations. The associations established from network are ignored.
- **SCTP Server Mode:**
  The M3UA PE doesn’t establish any SCTP associations. The associations are established from the network.
- **M3UA Client Mode:**
  The M3UA PE activates the M3UA ASPs.
- **M3UA Server Mode:**
  The M3UA PE does not activate the M3UA ASPs. The M3UA is activated from the network.

Abbreviations

ASP

Abstract Service Primitive

LKSCTP

Linux Kernel Stream Control Transmission Protocol

M3UA

MTP3 User Adaptation

M3UA PE

M3UA Protocol Emulation

MTP3

Message Transfer Part Level 3

PE

Protocol Emulation

SCTP

Stream Control Transmission Protocol

SS7

Signalling System 7

TTCN-3

Testing and Test Control Notation Version 3

References

[1] ETSI ES 201 873-1 V3.2.1
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] Installation Guide for TITAN TTCN-3 Test Executor

[5] TTCNv3 Requirement Specification for MSC R13

[6] M3UA Protocol Emulation for TTCN-3 Toolset, Function Specification

[7] RFC 2960
Stream Control Transmission Protocol (SCTP)

[8] RFC 3332
SS7 MTP3 User Adaptation Layer

[9] LKSCTP – An implementation of the SCTP in the Linux kernel
http://lksctp.sourceforge.net