

OsmoMSC - Bug #2856

No automatic testing of luCS interface

01/22/2018 01:33 AM - laforge

Status:	Resolved	Start date:	01/22/2018
Priority:	Normal	Due date:	
Assignee:	laforge	% Done:	100%
Category:	luCS support		
Target version:			
Resolution:			
Description			
<p>While we start to have some decent tests in our TTCN-3 based test suites for MGCP, BSSAP, MNCC, GSUP, etc., this all is 2G related testing so far.</p> <p>In order to test the 3G side of things, we should start with tests for luCS in OsmoMSC.</p> <p>As TITAN can not speak APER (aligned packed encoding rules) directly, we will have to do some kind of external transcoding. The process roughly looks like this:</p> <ul style="list-style-type: none">• TITAN parses the RANAP asn1 syntax and generates its own structured data types from it• TITAN can generate BER (or XER?) encoder/decoder from this• we need to hook up some transcoder that bidirectionally converts BER<->APER. This could be either an external program, or we could link it as a library via C++ code into TITAN <p>See https://www.eclipse.org/forums/index.php/t/1070344/ for a description of the problem and the usual approach to solve it. I'm attaching the key source file which contains the encoder/decoder functions.</p> <p>The converter could be generated either using our hacked version of asn1c (which we use in osmi-juh), or even using other (free or non-free) tools.</p>			
Related issues:			
Precedes OsmoSGSN - Bug #2857: No automatic testing of luPS interface		Stalled	01/23/2018 01/23/2018
Precedes OsmoHNBGW - Bug #2858: No automatic testing of luh interface		New	01/23/2018 01/23/2018

History

#1 - 01/22/2018 01:35 AM - laforge

- Precedes Bug #2857: No automatic testing of luPS interface added

#2 - 01/22/2018 01:38 AM - laforge

- Precedes Bug #2858: No automatic testing of luh interface added

#3 - 01/30/2018 04:38 PM - laforge

- % Done changed from 0 to 10

I've done some initial investigation + trials, and I could successfully compile the RANAP, RUA and HNBAP asn sources (taken from [wireshark.git/dissectors/epan/asn/](https://github.com/wireshark/wireshark/tree/master/dissectors/epan/asn/)) using `ttn3_makefilegen`. The result is some rather large C++ and even larger object files.

What's missing to complete the chain now is to link this code against `libosmo-ju` and the glue code for the BER<->APER transcoding

#4 - 05/30/2018 03:00 PM - laforge

- Tags set to `TTCN3, 3G`

#5 - 04/15/2019 09:50 AM - laforge

- Category set to luCS support
- Status changed from New to In Progress
- % Done changed from 10 to 20

I have RANAP encoding/decoding now working in TTCN-3, and I've also created a set of RANAP templates to use. Integration into the MSC test suite via RAN_Emulation.ttcn is currently ongoing.

#6 - 04/15/2019 08:17 PM - laforge

- Status changed from In Progress to Stalled

The core RANAP integration, the templates etc. appear to be working. The problem is that due to bugs in the proprietary asn1 compiler we're not able to reliably transcode between BER and APER at this point. setting to stalled again.

#7 - 04/20/2019 08:51 PM - laforge

- File iucs-auth-ttcn.pcap added
- Status changed from Stalled to In Progress
- % Done changed from 20 to 30

an update including related fixes of the asn1 compiler was released. I'm at the point where with very little hacks to MSC_Tests.ttcn I can get LU/auth dialogue between simulated RNC and OSmoMSC going, see attached pcap file.

It seems any fundamental ASN.1 related issues have been resolved, and libfftranscode is working as expected.

I'll clean up the code and work on more test integration shortly.

#8 - 04/20/2019 08:52 PM - laforge

some initial tests already uncovered some bugs, see <https://gerrit.osmocom.org/#/c/osmo-iuh/+13722/> and <https://gerrit.osmocom.org/#/c/osmo-iuh/+13721/>

#9 - 04/21/2019 10:34 AM - laforge

There are now packages of a working APER <-> BER transcoder available from <http://ftp.osmocom.org/binaries/libfftranscode/> - the packages are for debian9 x86_64, which is what we use in or docker containers.

#10 - 04/22/2019 08:24 PM - laforge

- % Done changed from 30 to 60

libfftranscode is integrated in our docker builds. On my system locally I've already been running a number of luCS TTCN3 tests. Will push them to gerrit once they have received more testing.

#11 - 05/05/2019 04:06 PM - laforge

- Status changed from In Progress to Resolved
- % Done changed from 60 to 100

MSC lu tests are now active in jenkins. Most of them fail due to [#3948](#), but this issue is about "not automatic testing" which is resolved, even if the testing results in 'fail'.

Files

CAM_EncDec.cc	6.28 KB	01/22/2018	laforge
iucs-auth-ttcn.pcap	1.1 KB	04/20/2019	laforge