

OsmoMSC - Bug #2856

No automatic testing of luCS interface

01/22/2018 01:33 AM - laforge

Status:	New	Start date:	01/22/2018
Priority:	Normal	Due date:	
Assignee:	laforge	% Done:	10%
Category:			
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-ih), or even using other (free or non-free) tools.</p>			
Related issues:			
Precedes OsmoSGSN - Bug #2857: No automatic testing of luPS interface		New	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/blob/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-ih` and the glue code for the BER<->APER transcoding

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

- Tags set to `TTCN3, 3G`

Files

