libosmo-sccp + libosmo-sigtran - Bug #2441

chopped-off pointcodes

08/15/2017 11:16 AM - dexter

Status:	Closed	**Start date:**	08/15/2017
Priority:	Urgent	**Due date:**	
Assignee:	laforge	**% Done:**	100%
Category:			
Target version:			
Spec Reference:			

Description

It seems that that the pointcode data is chopped off when receiving unittdata.

When looking at the attached trace.pcapng file, one can see that the RESET command is correctly transmitted, but the response, the RESET ACK is always sent to the wrong destination address. (187 instead of 2235). When converting those to numbers one can see that the addresses seem to be chopped off, presumably at the 8th bit:

```
2235 = 100010111011
187  =     10111011
```

When investigating further it turns out that the pointcode is already chopped off when the RESET is received:

```
Tue Aug 15 11:35:20 2017 <000a> a_iface.c:531 N-UNITDATA.ind(00 04 30 04 01 20 )
Tue Aug 15 11:35:20 2017 <000a> a_iface_bssap.c:184 Rx BSC UDT: 00 04 30 04 01 20
Tue Aug 15 11:35:20 2017 <000a> a_iface_bssap.c:157 Rx BSC UDT BSSMAP RESET
Tue Aug 15 11:35:20 2017 <000a> a_iface_bssap.c:110 Rx RESET from BSC RI=SSN_PC,PC=0.23.3,SSN=unkn
own 0xfe,GTI=NO_GT, sending RESET ACK
Tue Aug 15 11:35:20 2017 <000a> fsm.c:176 FSM RESET(FSM RESET INST)[0x55555589b7a0]{DISC}: Timeout
 of T0
Tue Aug 15 11:35:20 2017 <000a> a_reset.c:102 (RI=SSN_PC,PC=0.23.3,SSN=unknown 0xfe,GTI=NO_GT) res
et-ack timeout (T0) in state ST_DISC (disconnected), resending...
Tue Aug 15 11:35:20 2017 <000a> a_iface.c:443 Sending RESET to BSC RI=SSN_PC,PC=0.23.3,SSN=unknown
 0xfe,GTI=NO_GT
```

Presumably the upcoming primitive already contains the chopped pointcode.

Associated revisions

Revision c755c1d1 - 10/27/2017 02:36 PM - laforge

osmo_sccp_addr_encode(): Fix truncation of point codes

In osmo_sccp_addr_encode(), we accidentially truncated all point codes to 10 bits, where in reality we should have truncated them to 14 bits: One 'f' was missing in the bit-mask.

Closes: OS#2441
Change-Id: Iad67b674b5b5fd41996aa898a131e98900842dd8

Revision db736f43 - 10/27/2017 04:00 PM - laforge

implement unit tests for osmo_sccp_addr_{parse,encode}()

The recent bug with chopped-off point codes in SCCP Address handling has shown that this code could need proper test cases. This patch adds a testsuite for SCCP address encoding and decoding.

Related: OS#2441
Change-Id: I612352736ab33462ca0dd97798a2c437eadccb86

History

#1 - 10/27/2017 10:04 AM - laforge

- Priority changed from Normal to Urgent

#2 - 10/27/2017 12:01 PM - laforge

- Status changed from New to In Progress

- % Done changed from 0 to 30

it's likely a wrong mask in osmo_sccp_addr_encode(), where we only permit 0x3ff, i.e. 10 bit long point codes. Need to double-chech what's the actual length of point coes permitted in ITU SCCP and fix similar to the diff below:

```
@@ -237,7 +237,7 @@ int osmo_sccp_addr_encode(struct msgb *msg, const struct osmo_sccp_addr *in)

        if (in->presence & OSMO_SCCP_ADDR_T_PC) {
                sca->point_code_indicator = 1;
-               msgb_put_u16le(msg, in->pc & 0x3ff);
+               msgb_put_u16le(msg, in->pc & 0x3fff);
        }

        if (in->presence & OSMO_SCCP_ADDR_T_SSN) {
```

#3 - 10/27/2017 01:00 PM - laforge

Proposed fix in https://gerrit.osmocom.org/4445

#4 - 10/27/2017 03:49 PM - laforge

- Status changed from In Progress to Resolved

- % Done changed from 30 to 100

The fix has been merged.

I've meanwhile implemented some test cases that should help us avoiding any regressions about this in https://gerrit.osmocom.org/#/c/4449/

#5 - 02/06/2018 08:25 AM - laforge

- Status changed from Resolved to Closed

Files

trace.pcapng	5.68 KB	08/15/2017	dexter