OsmoSGSN - Bug #4824

vty comand "show sndcp" can cause SEGV in vty_dump_sne()

10/21/2020 03:11 PM - keith

Status:	New	**Start date:**	10/21/2020
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:			
Target version:			
Spec Reference:			

Description

```
(gdb) bt
#0  vty_dump_sne (vty=0x558339e52010, sne=0x558339e57360) at gprs_sndcp_vty.c:44
#1  0x000055833916afb7 in show_sndcp (self=0x55833939aac0 <show_sndcp_cmd>, vty=0x558339e52010, ar
gc=0, argv=0x7ffcb165cf50) at gprs_sndcp_vty.c:60
```

History

#1 - 10/21/2020 03:20 PM - keith

```
static void vty_dump_sne(struct vty *vty, struct gprs_sndcp_entity *sne)
{
    vty_out(vty, " TLLI %08x SAPI=%u NSAPI=%u:%s",
        sne->lle->llme->tlli, sne->lle->sapi, sne->nsapi, VTY_NEWLINE);
    vty_out(vty, "  Defrag: npdu=%u highest_seg=%u seg_have=0x%08x tot_len=%u%s",
        sne->defrag.npdu, sne->defrag.highest_seg, sne->defrag.seg_have,
        sne->defrag.tot_len, VTY_NEWLINE);
}
```

```
(gdb) p sne->lle->llme
$6 = (struct gprs_llc_llme *) 0x0
```

#2 - 10/21/2020 03:24 PM - keith

Can anyone shed any light on if this is an expected condition, and should be fixed by simply checking before access, of it it betrays an underlying bug?

#3 - 10/21/2020 04:16 PM - laforge

I don't think this is expected. The LLME is the LLC Management Entity. So you have LLC Enties with out its related management.

I can currently only find code that sets lle->llme during lle_init(), but I cannot find any other place (which would set it to NULL)

#4 - 10/21/2020 04:17 PM - laforge

maybe the entire 'lle' is gone, and the reference from the sne (sndcp entity) is already problematic? Can you print the lle in the debugger?

#5 - 10/21/2020 04:20 PM - laforge

also, the lle are an array inside the llme. So it's impossible that a LLE has no LLME. The back-pointer is not written to anywhere, so I think indeed the sne->lle pointer may be stale, or the entire sne.

#6 - 10/21/2020 04:26 PM - laforge

so the SNE is free'd via sndcp_sm_deactivate_ind(), which happens at PDP context termination.

the LLME with all its LLEs is free'd via gprs_llgmm_assign(..., new_tlli=TLLI_UNASSIGNED)

maybe there are other situations in whihc the LLME is freed, where the SNE is not free'd before? As all the gprs_llgmm_assign() calls originate from

gprs_gmm, it may be in the path for GGSN side PDP context termination from sgsn_libgtp.c -> sndcp_sm_deactivate_ind() where the SNDCP entity is not free'd.

I'll drop looking at it further here.

#7 - 12/30/2020 02:56 PM - lynxis

The rc3 setup has the same problem. But not always. maybe only when the llme was gone.

#8 - 12/30/2020 06:45 PM - lynxis

```
(gdb) bt
#0  vty_dump_sne (vty=0x55e91d42e840, sne=0x55e91d43a590) at ../../../../src/osmo-sgsn/src/sgsn/gprs_sndcp_vty
.c:44
#1  0x000055e91c3e1302 in show_sndcp (self=0x55e91c414b60 <show_sndcp_cmd>, vty=0x55e91d42e840, argc=0, argv=0
x7ffc22507bf0)
    at ../../../../src/osmo-sgsn/src/sgsn/gprs_sndcp_vty.c:60
#2  0x00007f6194d335b9 in cmd_execute_command_real (vty=vty@entry=0x55e91d42e840, cmd=cmd@entry=0x0, vline=<op
timized out>, vline=<optimized out>)
    at ../../../../src/libosmocore/src/vty/command.c:2602
#3  0x00007f6194d354ce in cmd_execute_command (vline=vline@entry=0x55e91d440600, vty=vty@entry=0x55e91d42e840,
 cmd=cmd@entry=0x0, vtysh=vtysh@entry=0)
    at ../../../../src/libosmocore/src/vty/command.c:2654
#4  0x00007f6194d37c1a in vty_command (vty=0x55e91d42e840) at ../../../../src/libosmocore/src/vty/vty.c:438
#5  vty_execute (vty=0x55e91d42e840) at ../../../../src/libosmocore/src/vty/vty.c:702
#6  vty_read (vty=<optimized out>) at ../../../../src/libosmocore/src/vty/vty.c:1428
#7  0x00007f6194d39ced in client_data (fd=0x55e91d433e98, what=1) at ../../../../src/libosmocore/src/vty/telne
t_interface.c:154
#8  0x00007f6194d01add in poll_disp_fds (n_fd=<optimized out>) at ../../../src/libosmocore/src/select.c:350
#9  _osmo_select_main (polling=<optimized out>) at ../../../src/libosmocore/src/select.c:378
#10 0x00007f6194d01b56 in osmo_select_main (polling=<optimized out>) at ../../../src/libosmocore/src/select.c:
417
#11 0x000055e91c3e6b38 in main (argc=3, argv=0x7ffc22508838) at ../../../../src/osmo-sgsn/src/sgsn/sgsn_main.c
:532
(gdb) frame 0
#0  vty_dump_sne (vty=0x55e91d42e840, sne=0x55e91d43a590) at ../../../../src/osmo-sgsn/src/sgsn/gprs_sndcp_vty
.c:44
44              vty_out(vty, " TLLI %08x SAPI=%u NSAPI=%u:%s",
(gdb) p locals
No symbol "locals" in current context.
(gdb) info locals
No locals.
(gdb) info local
No locals.
(gdb) info locals
No locals.
(gdb) p *sne
$1 = {list = {next = 0x55e91c414b40 <gprs_sndcp_entities>, prev = 0x55e91d4348e0}, ra_id = {mcc = 262, mnc = 4
2, mnc_3_digits = false, lac = 23, rac = 0 '\000'},
  lle = 0x55e91d43b988, nsapi = 5 '\005', tx_npdu_nr = 195, rx_state = SNDCP_RX_S_FIRST, defrag = {npdu = 197,
 highest_seg = 2 '\002', seg_have = 7, no_more = 1,
    tot_len = 1159, frag_list = {next = 0x55e91d43a5d0, prev = 0x55e91d43a5d0}, timer = {node = {rb_parent_col
or = 0, rb_right = 0x0, rb_left = 0x0}, list = {next = 0x0,
        prev = 0x0}, timeout = {tv_sec = 0, tv_usec = 0}, active = 0, cb = 0x0, data = 0x55e91d43a590}, pcomp
= 0 '\000', dcomp = 0 '\000', proto = 0x55e91d439800,
    data = 0x55e91d435840}}

(gdb) p *sne->lle
$3 = {list = {next = 0x0, prev = 0x0}, sapi = 0, llme = 0x0, state = 0, t200 = {node = {rb_parent_color = 0, r
b_right = 0x0, rb_left = 0x0}, list = {next = 0x0,
      prev = 0x0}, timeout = {tv_sec = 0, tv_usec = 0}, active = 0, cb = 0x0, data = 0x0}, t201 = {node = {rb_
parent_color = 0, rb_right = 0x0, rb_left = 0x0}, list = {
      next = 0x0, prev = 0x0}, timeout = {tv_sec = 0, tv_usec = 0}, active = 0, cb = 0x0, data = 0x0}, v_sent
= 0, v_ack = 0, v_recv = 0, vu_send = 0, vu_recv = 0,
  vu_recv_last = 0, vu_recv_duplicates = 0, oc_i_send = 0, oc_i_recv = 0, oc_ui_send = 0, oc_ui_recv = 0, retr
ans_ctr = 0, params = {iov_i_exp = 0, t200_201 = 0, n200 = 0,
    n201_u = 0, n201_i = 0, mD = 0, mU = 0, kD = 0, kU = 0}, xid = 0x0}
```

#9 - 12/31/2020 01:16 AM - lynxis

it happens when moving from 2G to 3G!