

OsmoPCU - Bug #3013

regression: GPRS fatally unresponsive since commit 'Rewrite Packet Downlink Assignment'

02/27/2018 08:47 PM - neels

Status: Resolved	Start date: 02/27/2018
Priority: Normal	Due date:
Assignee: neels	% Done: 100%
Category:	
Target version:	
Spec Reference:	
Description	
<p>While testing code changes based on current osmo-pcu master, I have noticed severe service outage, symptom from the user experience: the remote hosts not responding. At first a web page may load, but soon after, loading any other pages will completely stop working -- the downlink completely stops for the remaining lifetime of the PDP Context.</p> <p>Apparently receiving on GPRS_NS a FLOW-CONTROL-BVC + FLOW-CONTROL-BVC-ACK pair triggers the behavior, but that's just a hunch.</p> <p>I have tried an earlier osmo-pcu version which not exhibiting this behavior, and bisected the failure down to:</p> <pre>commit 896574e92bea09ed8d39688b6fdf504e84521746 Author: Max <msuraev@systemocom.de> Date: Tue Jan 9 18:45:41 2018 +0100 Rewrite Packet Downlink Assignment Use bitvec_set_*() directly without external write pointer tracking to simplify the code. This is part of IA Rest Octets (3GPP TS 44.018 §10.5.2.16) which is the last part of the message so it should not interfere with the rest of encoding functions. The tests are adjusted accordingly. Change-Id: I52ec9b07413daabba8cd5f1fba5c7b3af6a33389 Related: OS#1526</pre>	
Related issues:	
Related to OsmoPCU - Feature #3014: fix re-apply patches reverted by #3013, r...	Resolved 02/27/2018

History

#1 - 02/27/2018 09:13 PM - neels

Tried to revert the commit in question (with some conflict resolution) but it's still broken after that.

Note that the immediate parent commit of above regression is "Rewrite EGPRS Packet Uplink Assignment", which does sound similar. I haven't tested EGPRS.

I also notice that the commit in question says the rationale is to "simplify the code", yet the test expectations are modified along with it, particularly message octets. I would have expected code refactoring to not yield any PDU changes.

Short of understanding what exactly is going wrong, it seems that we need to "start over" from 2141962baf95bfaf11f19dacd59f7b8ac8d49ca3, cherry-picking commits that seem independent from the regression, and see if we can get osmo-pcu stable again that way. After that, we can re-evaluate the commits introducing the regression.

Unless of course someone is apt enough to fully understand the failure right now.

#2 - 02/27/2018 09:55 PM - neels

I've found a reasonably small set of commits to revert painlessly that renders osmo-pcu usable again:

https://gerit.osmocom.org/#/q/status:open+project:osmo-pcu+branch:master+topic:fix_regression

=

- <https://gerrit.osmocom.org/6976> Revert "Use Timing Advance Index in UL assignments"
- <https://gerrit.osmocom.org/6977> Revert "Rewrite Packet Uplink Assignment"
- <https://gerrit.osmocom.org/6978> Revert "Rewrite Packet Downlink Assignment"
- <https://gerrit.osmocom.org/6979> Revert "Rewrite EGPRS Packet Uplink Assignment"

I'm creating a new ticket that asks for re-adding these patches: [#3014](#) ... and (almost) closing this one.

#3 - 02/27/2018 09:56 PM - neels

- Status changed from New to In Progress
- Assignee set to neels
- % Done changed from 0 to 90

#4 - 02/27/2018 10:05 PM - neels

- Related to Feature #3014: fix re-apply patches reverted by #3013, related: UL and DL packet assignment, and Timing Advance added

#5 - 02/28/2018 12:17 AM - neels

- File os3013_gprs_works__bts_master__pcu_neels-fix_regression-414fcb0.pcapng added
- File os3013_gprs_completely_unusable_1__bts_master__pcu_master_0.4.0.97-731e.pcapng added
- File os3013_gprs_almost_completely_unusable_2__bts_master__pcu_master_0.4.0.97-731e.pcapng added

attaching pcaps.

Enabling too much GSMTAP made the PCU unusable, I have in the end enabled these few. I hope it is sufficient. Let me know if specific logs / gsmtaps should be added and I can easily do new traces.

#6 - 02/28/2018 09:37 AM - laforge

Enabling too much GSMTAP made the PCU unusable

Yes, that's expected and this is why running OsmoPCU on sysmobts-1xxx for R&D is not the best possible setup. There's simply not a lot of spare CPU cycles for additional debugging/logging in the code.

I suggest PCU development/debugging is primarily done on a different hardware platform, Either with osmo-bts-trx + osmo-pcu on a normal x86 PC, or e.g. on a sysmobts-2100 which has **much** more CPU and a PHY very similar to the 1002.

#7 - 04/11/2018 12:28 PM - neels

- Status changed from In Progress to Resolved
- % Done changed from 90 to 100

The reverts are merged, checking why the patches fail and re-committing them is tracked in [#3014](#)

Files

os3013_gprs_works__bts_master__pcu_neels-fix_regression-414fcb0.pcapng	2.31 MB	02/28/2018	neels
os3013_gprs_completely_unusable_1__bts_master__pcu_master_0.4.0.97-731e.pcapng	2.09 MB	02/28/2018	neels
os3013_gprs_almost_completely_unusable_2__bts_master__pcu_master_0.4.0.97-731e.pcapng	4.16 MB	02/28/2018	neels