

OsmoGSMTester - Bug #4151

osmo-gsm-tester: osmo-trx-lms process sometimes kept forever in zombie-like state after killing it

08/14/2019 09:56 AM - pespin

Status:	Stalled	Start date:	08/14/2019
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:			
Target version:			
Spec Reference:			
Description			
It was spotted several times that all osmo-trx-lms tests in osmo-gsm-tester fail with message:			
<pre>socket.c:367 unable to bind socket:10.42.42.117:4237: Address already in use</pre>			
Close lookup shows osmo-trx-lms stil running but idle (not consuming CPU):			
<pre># ps -ef grep osmo-trx-lms root 14643 14604 0 11:47 pts/1 00:00:00 grep osmo-trx-lms jenkins 55210 1 0 Aug06 ? 00:00:53 /osmo-gsm-tester-trx/last_run/osmo-trx/bin/osmo-tr x-lms -C /osmo-gsm-tester-trx/last_run/osmo-trx.cfg</pre>			
In order to get process creation time (too see which test caused the issue and if logs around it provide more information):			
<pre># ls -ld --time-style=full-iso /proc/\$(pidof osmo-trx-lms) dr-xr-xr-x 9 jenkins jenkins 0 2019-08-06 10:37:14.274166715 +0200 /proc/55210</pre>			
At that time, following run was in place: https://jenkins.osmocom.org/jenkins/view/osmo-gsm-tester/job/osmo-gsm-tester_run-prod/1926/			
And the test: trial-1926 gprs:trx-lms+mod-bts0-numtrx2+mod-bts0-chanallocdescend cs_paging_gprs_active.py			
The test runs and at some fails (expected since multi-trx is not yet supported in osmo-trx-lms) and then osmo-gsm-tester goes over regular procedure to kill all processes (in the case of osmo-trx-lms, it kills the ssh client, which should end up killing its child through the script handler):			
<pre>10:38:00.558825 --- ParallelTerminationStrategy: DBG: Scheduled to terminate 22 processes. [process.py:108] 10:38:00.560001 --- ParallelTerminationStrategy: DBG: Starting to kill with SIGTERM [process .py:116] ... 10:38:00.669914 run osmo-trx-lms(pid=1883): Terminating (SIGTERM) [trial-1926↪gprs:trx- lms+mod-bts0-numtrx2+mod-bts0-chanallocdescend↪osmo-bts-trx↪osmo-trx-lms↪osmo-trx-lms(pid=1883)] [process.py:236] ... 10:38:00.773158 --- ParallelTerminationStrategy: PID 1883 died... [process.py:75] 10:38:00.773706 run osmo-trx-lms(pid=1883): DBG: Cleanup [trial-1926↪gprs:trx-lms+mod-b ts0-numtrx2+mod-bts0-chanallocdescend↪osmo-bts-trx↪osmo-trx-lms↪osmo-trx-lms(pid=1883)] [process. py:265] 10:38:00.776101 run osmo-trx-lms(pid=1883): Terminated {rc=36608} [trial-1926↪gprs:trx- lms+mod-bts0-numtrx2+mod-bts0-chanallocdescend↪osmo-bts-trx↪osmo-trx-lms↪osmo-trx-lms(pid=1883)] [process.py:270]</pre>			
So my guess is not that ssh killing its child process is not working, but rather than when running with multi-trx we may end up in some race condition which somehow blocks osmo-trx-lms and prevents it from exiting.			
Related issues:			
Related to OsmoTRX - Bug #3346: osmo-trx-lms: Multi channel support: "R_CTL_L...		Rejected	06/13/2018

History

#1 - 08/14/2019 10:35 AM - pespin

- Status changed from New to Feedback

I submitted a couple of commits to gerrit to help debug/workaround the issue:

remote: <https://gerrit.osmocom.org/c/osmo-gsm-tester/+15190> bts-trx: Improve logging and trap SIGTERM in ssh_sigkiller.sh

remote: <https://gerrit.osmocom.org/c/osmo-gsm-tester/+15191> default-suites: Drop multi-trx osmo-trx-lms tests

Let's see how it behaves with those two applied, if the issue still shows up then.

#2 - 08/14/2019 10:36 AM - pespin

- Description updated

#3 - 08/14/2019 11:02 AM - pespin

Manually killing the process (kill \$PID) doesn't work, the process is really stuck.

gdb attaching:

```
Attaching to process 55210
[New LWP 55211]
[New LWP 55212]
[New LWP 55213]
```

```
warning: Could not load vsyscall page because no executable was specified
0x000007f8d5f37ef7c in ?? ()
(gdb) thread apply all bt
```

```
Thread 4 (LWP 55213):
#0  0x000007f8d5d0c48bd in ?? ()
#1  0x0000000000000000 in ?? ()
```

```
Thread 3 (LWP 55212):
#0  0x000007f8d5f37ef7c in ?? ()
#1  0x0000000000000000 in ?? ()
```

```
Thread 2 (LWP 55211):
#0  0x000007f8d5d0c48bd in ?? ()
#1  0x0000000000000000 in ?? ()
```

```
Thread 1 (LWP 55210):
#0  0x000007f8d5f37ef7c in ?? ()
#1  0x0000000000000000 in ?? ()
```

Looking at maps:

```
Thread1 and Thread3: 7f8d5f36f000-7f8d5f387000 r-xp 00000000 08:02 11666510          /lib/x86_64-linux-gnu/libpthread-2.24.so
7F8D5F37EF7C-7f8d5f36f000 = FF7C
```

```
# addr2line -a -p -C -f -i -e /lib/x86_64-linux-gnu/libpthread-2.24.so 0xff7c
0x000000000000ff7c: __lll_lock_wait at /build/glibc-77giwP/glibc-2.24/nptl/./sysdeps/unix/sysv/linux/x86_64/lowlevellock.S:135
```

```
Thread2 and Thread4: 7f8d5cfe5000-7f8d5d17a000 r-xp 00000000 08:02 11666495          /lib/x86_64-linux-gnu/libc-2.24.so
```

```
# addr2line -a -p -C -f -i -e /lib/x86_64-linux-gnu/libc-2.24.so DF8BD
0x000000000000df8bd: __poll at /build/glibc-77giwP/glibc-2.24/io/./sysdeps/unix/syscall-template.S:84
```

So 2 threads waiting in poll and 2 in a lock...

Backtrace and whatever is not working properly because the binary + osmocom libraries were re-copied from later tests in the same directory.

#4 - 08/14/2019 11:03 AM - pespin

- Related to Bug #3346: osmo-trx-lms: Multi channel support: "R_CTL_LPF range limit reached" added

#5 - 08/16/2019 03:51 PM - pespin

After removing the multi-trx tests for LMS it there's no zombie osmo-trx-lms anymore.
Let's keep the issue open to re-check again once we support multiTRX in osmo-trx-lms.

#6 - 09/17/2019 11:22 AM - pespin

- *Status changed from Feedback to Stalled*

#7 - 09/30/2020 08:08 PM - laforge

- *Assignee deleted (pespin)*