

### 3.0基站产品测试 - 错误 #4743

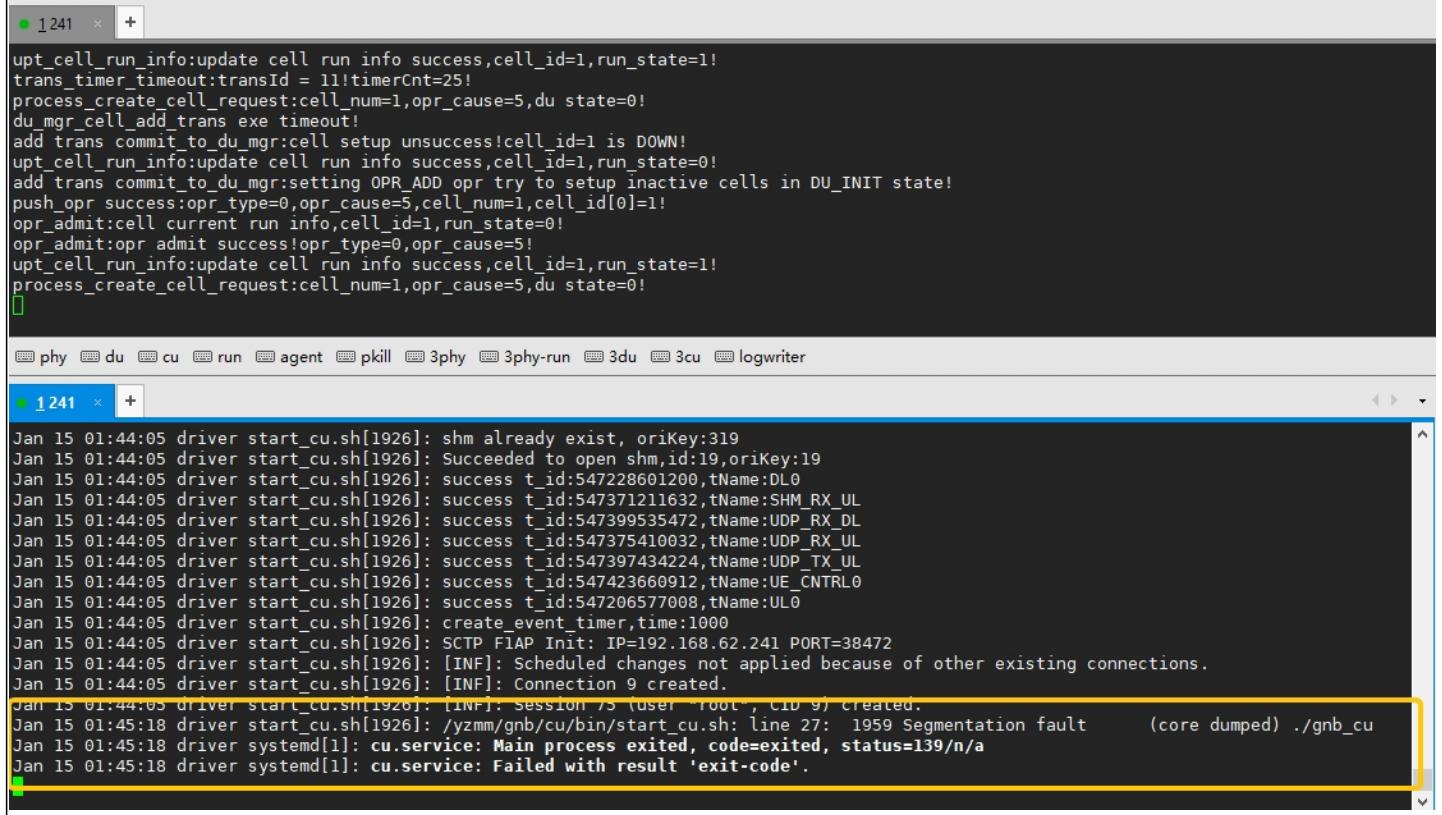
#### 西安241基站从Rel\_3.1.4\_Pre1T4升级到Rel\_3.1.5\_Pre1T1版本启动基站时cu必现挂死

2026-01-15 10:20 - 孙浩

状态:	已解决	开始日期:	2026-01-15
优先级:	一般	计划完成日期:	
指派给:	孙浩	% 完成:	0%
类别:		预期时间:	0.00 小时
目标版本:		耗时:	0.00 小时
问题归属:	CU	目标解决问题版本:	Rel_3.1.5
发现问题版本:	Rel_3.1.5		

#### 描述

【问题描述】西安241基站从Rel\_3.1.4\_Pre1T4升级到Rel\_3.1.5\_Pre1T1版本，启动基站就cu挂死；



```
upt_cell_run_info:update cell run info success,cell_id=1,run_state=1!
trans_timer_timeout:transId = 11!timerCnt=25!
process_create_cell_request:cell_num=1,opr_cause=5,du state=0!
du mgr_cell_add_trans exe timeout!
add trans commit_to_du_mgr:cell setup unsuccess!cell_id=1 is DOWN!
upt_cell_run_info:update cell run info success,cell_id=1,run_state=0!
add_trans commit_to_du_mgr:setting OPR_ADD opr try to setup inactive cells in DU_INIT state!
push_opr success:opr_type=0,opr_cause=5,cell_num=1,cell_id[0]=1!
opr_admit:cell current run info,cell_id=1,run_state=0!
opr_admit:opr admit success!opr_type=0,opr_cause=5!
upt_cell_run_info:update cell run info success,cell_id=1,run_state=1!
process_create_cell_request:cell_num=1,opr_cause=5,du state=0!

```

```
Jan 15 01:44:05 driver start_cu.sh[1926]: shm already exist, oriKey:319
Jan 15 01:44:05 driver start_cu.sh[1926]: Succeeded to open shm,id:19,oriKey:19
Jan 15 01:44:05 driver start_cu.sh[1926]: success t_id:547228601200,tName:DLO
Jan 15 01:44:05 driver start_cu.sh[1926]: success t_id:547371211632,tName:SHM_RX_UL
Jan 15 01:44:05 driver start_cu.sh[1926]: success t_id:547399535472,tName:UDP_RX_DL
Jan 15 01:44:05 driver start_cu.sh[1926]: success t_id:547375410032,tName:UDP_RX_UL
Jan 15 01:44:05 driver start_cu.sh[1926]: success t_id:547397434224,tName:UDP_TX_UL
Jan 15 01:44:05 driver start_cu.sh[1926]: success t_id:547423660912,tName:UE_CNTROL0
Jan 15 01:44:05 driver start_cu.sh[1926]: success t_id:547206577008,tName:UL0
Jan 15 01:44:05 driver start_cu.sh[1926]: create_event_timer,time:1000
Jan 15 01:44:05 driver start_cu.sh[1926]: SCTP FLAP Init: IP=192.168.62.241 PORT=38472
Jan 15 01:44:05 driver start_cu.sh[1926]: [INF]: Scheduled changes not applied because of other existing connections.
Jan 15 01:44:05 driver start_cu.sh[1926]: [INF]: Connection 9 created.
Jan 15 01:44:05 driver start_cu.sh[1926]: [INF]: session /s (user `root` , cu 9) created.
Jan 15 01:45:18 driver start_cu.sh[1926]: /yzmm/gnb/cu/bin/start_cu.sh: line 27: 1959 Segmentation fault      (core dumped) ./gnb_cu
Jan 15 01:45:18 driver systemd[1]: cu.service: Main process exited, code=exited, status=139/n/a
Jan 15 01:45:18 driver systemd[1]: cu.service: Failed with result 'exit-code'.
```

#### 历史记录

#1 - 2026-01-15 10:23 - 孙浩

- 主题从 西安241基站从Rel\_3.1.4\_Pre1T4升级到Rel\_3.1.5\_Pre1T1版本启动基站就cu挂死 变更为  
西安241基站从Rel\_3.1.4\_Pre1T4升级到Rel\_3.1.5\_Pre1T1版本启动基站cu偶现挂死

#2 - 2026-01-15 15:10 - 孙浩

- 主题从 西安241基站从Rel\_3.1.4\_Pre1T4升级到Rel\_3.1.5\_Pre1T1版本启动基站cu偶现挂死 变更为  
西安241基站从Rel\_3.1.4\_Pre1T4升级到Rel\_3.1.5\_Pre1T1版本启动基站cu必现挂死

#3 - 2026-01-15 15:10 - 孙浩

- 主题从 西安241基站从Rel\_3.1.4\_Pre1T4升级到Rel\_3.1.5\_Pre1T1版本启动基站cu必现挂死 变更为  
西安241基站从Rel\_3.1.4\_Pre1T4升级到Rel\_3.1.5\_Pre1T1版本启动基站时cu必现挂死

#4 - 2026-01-16 14:06 - 杨杨乐

- 状态从 新建 变更为 进行中

#### 【问题原因】

1.问题出现在241基站上，241基站配置了一个邻区，邻区IP为:192.168.62.242  
 2.基站192.168.62.242的机器启动了，但是cu一直没有启动  
 3.基站启动后会首先ping192.168.62.242，可以ping通  
 4.基站会去连接192.168.62.242，但是由于242的CU没启动，sctp连接失败了；失败后将241的xn客户端的套接字close了,该套接字的数字比如为：9  
 5.随后基站241调用accept接入F1的客户端，代码如下,accept接收到了F1的客户端newfd也是9

```

int32_t newfd = -1;
if ((newfd = accept(client_fd, NULL, 0)) != -1) // newfd为客户端连接的套接字 {
//不为-1，代表有xn或f1的客户端接入
ngp::sctp_sock_apis* tmp = it_fd->second;
if (ret_t::FAILURE == epoll_register_fd(newfd, EPOLLIN, tmp)) {
close(newfd);
continue;
}
}
6.由于之前XN客户端的套接字:9,已经存入了map容器m_all_fd_map中，但是在xn客户端连接失败时没有从容器中移除  

7.f1客户端端接入后，套接字还是9，调用epoll_register_fd时，会判断在m_all_fd_map中已经存在，执行epoll_ctl的动作变成了EPOLL_CTL_MOD；由于之前xn客户端使用9时链接xn服务端失败时调用了close；  

调用close会导致m_epoll_fd的监控socket中移除该套接字。导致f1客户端接入时无法完成epoll_ctl的Mod动作，因为epoll_ctl不允许不增加就Mod，然后就释放了f1客户端  

8.然后f1的客户端就会在短时间内频繁的建立和断开连接，发送大量的F1SetupRequest，同时cu无法在短时间内处理这么多F1SetupRequest；然后下发了F1SetupFailure  

9，CU会发送F1SetupFailure，会将gnb_mgr_du对象中的指针ongoing_msg置为空指针；在之后又一次收到F1SetupRequest后，会使用ongoing_msg，但是没有判断，导致程序崩溃
```

**【修改方案】**

- 1.在xn客户端连接xn服务器时，如果失败了则从m\_all\_fd\_map中移除该套接字
- 2.在使用ongoing\_msg前做有效性判断

#### 【回归方法和注意事项】

- 1.配置一个开机的邻基站，但是不启动三层
- 2.多次开机，关注是否能正常启动，如果可以则关闭该问题单

该版本代码在Rel\_3.1.5\_Pre1T2已合入，在Rel\_3.1.5\_Pre1T2版本验证

#5 - 2026-01-16 14:06 - 杨杨乐

- 状态从进行中变更为审视

#6 - 2026-01-16 14:07 - 杨杨乐

- 状态从审视变更为转测试

- 指派给从杨杨乐变更为孙浩

#7 - 2026-01-19 10:11 - 孙浩

- 状态从转测试变更为已解决

Rel\_3.1.5\_Pre1T2版本基站正常启动多次未出现挂死，问题解决。

#### 文件

cu启动挂死截图.jpg	350 KB	2026-01-15	孙浩
--------------	--------	------------	----