All channels are ending up in state BORKEN shortly after bringing up the BTS.
Will add any more info to the ticket as it becomes available.

Log of transition to BORKEN:

Mar 26 13:53:48 tosepan osmo-bsc[12799]: DRLL DEBUG <0000> lchan_select.c:91 looking for lchan SDCCH: (bts=0,trx=1,ts=1,pchan=SDCCH8,state=IN_USE) ss=3 is available
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DRLL DEBUG <0000> lchan_select.c:258 (bts=0) lchan_select_by_type(SDCCH)
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN INFO <000f> lchan_select.c:262 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](UNUSED): (type=SDCCH) Selected
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> abis_rsl.c:1160 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](UNUSED): (type=SDCCH) MS: Channel Request: reason=LOCATION_UPDATE ra=0x02 ta=65
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> lchan_fsm.c:329 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](UNUSED): Received Event LCHAN_EV_ACTIVATE
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> lchan_fsm.c:553 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](UNUSED): state_chg to WAIT_TS_READY
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> gsm_data.c:868 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](WAIT_TS_READY): (type=SDCCH) MS Power level update requested: 33 dBm
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> gsm_data.c:900 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](WAIT_TS_READY): (type=SDCCH) MS Power level update (power class 0): 0
  -> 5
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN INFO <000f> lchan_fsm.c:636 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](WAIT_TS_READY): (type=SDCCH) Activation requested: FOR_MS_CHANNEL_REQ UEST voice=no MGW-ci=none type=SDCCH tch-mode=SIGNALLING encr-alg=A5/0 ck=none
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DTS DEBUG <0010> lchan_fsm.c:640 timeslot(0-1-1-SDCCH8)[0x55d1ebcfbf60](UNUSED): Received Event TS_EV_LCHAN_REQUESTED
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> timeslot_fsm.c:639 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](WAIT_TS_READY): Received Event TS_EV_LCHAN_REQUESTED
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> lchan_fsm.c:654 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](WAIT_TS_READY): state_chg to WAIT_ACTIV_ACK
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DRSL DEBUG <0003> abis_rsl.c:530 (bts=0,trx=1,ts=1,pchan=SDCCH8,state=IN_USE) Tx RSL Channel Activate with act_type=INITIAL
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DLMJ DEBUG <0019> input/eld.c:112 TX: 08 21 01 59 03 00 0 6 04 00 03 01 00 05 06 64 59 e0 fa 72 00 04 00 0d 05 18 41
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DLMJ DEBUG <0019> e1_input.c:729 E1TS(0:1) <= len = 4, sapi(62) tei(0)
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DLMJ DEBUG <0019> el_input.c:729 E1TS(0:4) <= len = 4, sapi(0) tei(1)
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DLMJ DEBUG <0019> el_input.c:729 E1TS(0:4) <= len = 11, sapi(0) tei(1)
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DLMJ DEBUG <0019> el_input.c:769 E1TS(0:4) RX: 08 23 01 5 9 1a 01 2f sapi=0 tei=1
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> abis_rsl.c:1232 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](WAIT_ACTIV_ACK): (type=SDCCH) RX CHANNEL ACTIVATE NACK
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN ERROR <000f> abis_rsl.c:961 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](WAIT_ACTIV_ACK): (type=SDCCH) CHANNEL ACTIVATE NACK (cause=Resource not available, unspecified [ 2f ])
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> abis_rsl.c:966 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810](WAIT_ACTIV_ACK): Received Event LCHAN_EV_RSL_CHAN_ACTIV_NACK
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> lchan_fsm.c:740 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810]{WAIT_ACTIV_ACK}: (type=SDCCH) Handling failure, will then transition to state BORK
EN
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN ERROR <000f> lchan_fsm.c:81 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810]{WAIT_ACTIV_ACK}: (type=SDCCH) lchan allocation failed in state WAIT_ACTIV_ACK: Chan Activ NACK: Resource not available, unspecified (0x2f)
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN NOTICE <000f> lchan_fsm.c:106 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810]{WAIT_ACTIV_ACK}: (type=SDCCH) Tx Immediate Assignment Reject (lchan allocation failed in state WAIT_ACTIV_ACK: Chan Activ NACK: Resource not available, unspecified (0x2f))

Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> lchan_fsm.c:740 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810]{WAIT_ACTIV_ACK}: state_chg to BORKEN
Mar 26 13:53:48 tosepan osmo-bsc[12799]: DCHAN DEBUG <000f> lchan_fsm.c:382 lchan(0-1-1-SDCCH8-3)[0x55d1ebd0d810]{BORKEN}: (type=SDCCH) Clearing lchan state

Related issues:
Related to OsmoBSC - Feature #5106: Watchdog that would try to un-BORKE BORKe...

Associated revisions
Revision 1823f89c - 04/04/2021 01:39 PM - keith
Ignore CHANnel ReQuireD with Access Delay IE > 63

It is observed that a CHANnel ReQuireD with access delay greater than 63 can be received from the Ericsson RBS. This results in osmo-bsc sending back a CHANnel ACTIVation with a Timing Advance IE containing the access delay value. The RBS NACKs this, leading to a BORKEN Channel.

This patch makes the maximum acceptable access delay vty-configurable and Ignores CHANnel ReQuireD RSL Messages with Access Delay IE greater than that configured. Default value is 63.

Related: OS#5096
Change-Id: Ie8987bcc0e43921bc753162b77a0efc68799b3ce

History
#1 - 03/26/2021 10:39 PM - keith
- File bsc-e1.pcap added

bsc-e1.pcap starts shortly after bring up, to all chans being in BORKEN state

Nack filter: gsm_abis_rsl.msg_type == 0x23

#2 - 03/26/2021 10:40 PM - keith
- File osmo-bsc.cfg added

osmo-bsc Config.

#3 - 03/27/2021 10:35 PM - keith

For what it may be worth, simply transitioning to WAIT_RF_RELEASE_ACK instead of BORKEN does seem to eventually return the state to UNUSED and allows further use of the channels.

--- a/src/osmo-bsc/lchan_fsm.c
+++ b/src/osmo-bsc/lchan_fsm.c
@@ -730,7 +730,7 @@ static void lchan_fsm_wait_activ_ack(struct osmo_fsm_inst *fi, uint32_t event, v
 lchan->release.rr_cause = bsc_gsm48_rr_cause_from_rsl_cause(lchan->release.rsl_error_c
 ause);
     if (lchan->release.rsl_error_cause != RSL_ERR_RCH_ALR_ACTV_ALLOC)
-          next_state = LCHAN_ST_BORKEN;
+          next_state = LCHAN_ST_WAIT_RF_RELEASE_ACK;
     else
         /* Taking this over from legacy code: send an RF Chan Release even though
            the Activ was NACKed. Is this really correct? */
I had a look at the capture file, and I think I know what happens:

- from time to time the BSC gets CHANnel ReQuireD with Access Delay 65 (!)
  - the usual range is 0 .. 63, and 65 * 550 m == 35.8 km is quite far...
  - AFAIR, values exceeding this range are allowed for GSM-450 only
- osmo-bsc tries to allocate a channel by sending RSL CHANnel ACTIVation message
  - this message contains the Timing Advance IE with the expected delay
  - apparently, your BTS is not happy about such a huge delay
- the BTS rejects RSL CHANnel ACTIVation by sending NACK
- osmo-bsc gets confused and treats the timeslot as 'BROKEN'

In osmo-bts we have configurable filtering for the delay / quality of the Access and Normal bursts, so 'outsiders' never reach the BSC. And I guess you need to configure / implement something similar for your Ericsson BTS. I have no idea if this BTS itself can do filtering, but in the worst case we can implement it in osmo-bsc.

Feel free to cherry-pick this patch and give it a try:

https://git.osmocom.org/osmo-bsc/commit/?h=fixeria/rach&id=f2d3cdddef982c0c8592b9b4e5ef4089a1910ea3c

#5 - 03/28/2021 07:34 PM - keith
- Status changed from Feedback to In Progress

Nice, I tested your patch adding a log line and certainly can see it happening from time to time, however, changing the MNC and bringing up the network again with 1,000s of phones then attempting to LUR, I still end up with most if not all BORKEN SDCCH after a few minutes.

What other information would help?

#6 - 03/28/2021 07:50 PM - laforge

fixeria wrote:

I had a look at the capture file, and I think I know what happens:

thanks for the debugging.

- from time to time the BSC gets CHANnel ReQuireD with Access Delay 65 (!)
  - the usual range is 0 .. 63, and 65 * 550 m == 35.8 km is quite far...
  - AFAIR, values exceeding this range are allowed for GSM-450 only
- osmo-bsc tries to allocate a channel by sending RSL CHANnel ACTIVation message
  - this message contains the Timing Advance IE with the expected delay
  - apparently, your BTS is not happy about such a huge delay

I think the assignment of “extended range” timing advance is something that likely requires additional [proprietary] siganling. We don’t need to bother as we don’t want to implement it.

- the BTS rejects RSL CHANnel ACTIVation by sending NACK
- osmo-bsc gets confused and treats the timeslot as 'BROKEN'

osmo-bsc should for sure not get confused by a CHAN ACT NACK, is there a separate bug here?

In osmo-bts we have configurable filtering for the delay / quality of the Access and Normal bursts, so 'outsiders' never reach the BSC. And I guess you need to configure / implement something similar for your Ericsson BTS. I have no idea if this BTS itself can do filtering, but in the worst case we can implement it in osmo-bsc.

I think osmo-bsc should do a proper check, let’s call it "input validation". It should be done generically so it will work for any type of BTS.

#7 - 03/28/2021 07:57 PM - laforge

keith wrote:

What other information would help?
if this fix worked as expected, the log+pcap should now look quite different. please update, thanks!

#8 - 03/28/2021 07:58 PM - keith
- File lur-bomb.pcap added

So looking at the original pcap again, I see that, as you rightly spotted, all those Nacks are indeed the results of TA > 63 - so looks like that patch is a runner. Should I prepare this, with associated vty config? rach-max-ta or some such?

Adding a pcap of bringing the network up with a new MNC..

I left it run a little longer just now, and interestingly it's only on TRX 0 that everything gets b0rked:

```
OsmoBSC# show lchan summary
BTS 0, TRX 0, Timeslot 1 SDCCH8, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 1 SDCCH8, Lchan 1, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 1 SDCCH8, Lchan 2, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 1 SDCCH8, Lchan 3, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 1 SDCCH8, Lchan 4, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 1 SDCCH8, Lchan 5, Type SDCCH, State WAIT_ACTIV_ACK - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 1 SDCCH8, Lchan 6, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 1 SDCCH8, Lchan 7, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 2 SDCCH8, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 2 SDCCH8, Lchan 1, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 2 SDCCH8, Lchan 2, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 2 SDCCH8, Lchan 3, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 2 SDCCH8, Lchan 4, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 2 SDCCH8, Lchan 5, Type SDCCH, State WAIT_RLL_RTP_ESTABLISH - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 2 SDCCH8, Lchan 6, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 2 SDCCH8, Lchan 7, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 3 TCH/F, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 3 TCH/F, Lchan 1, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 3 TCH/F, Lchan 2, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 4 TCH/F, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 4 TCH/F, Lchan 1, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 5 TCH/F, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 6 TCH/F, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 6 TCH/F, Lchan 1, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 7 TCH/F, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 7 TCH/F, Lchan 1, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 8 TCH/F, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 8 TCH/F, Lchan 1, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 9 TCH/F, Lchan 0, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
BTS 0, TRX 0, Timeslot 9 TCH/F, Lchan 1, Type NONE, State BORKEN - L1 MS Power: 0 dBm RXL-FULL-dl: -110 dBm RXL-FULL-ul: -110 dBm
```

05/23/2021
I also see chans actually leaving the and re-entering the borken state, which I thought was not possible. I guess full osmo-bsc log would be a useful thing here. I'm not sure how to deal with the lack of gsmtap logs embedded in the pcap.

Adding a DEBUG level osmo-bsc.log, Maybe grepping on it might help. I just let it run until it was about 100M

[Note this log does not correspond directly to lur-bomb.pcap]

This time it happens because of a different problem:

DCHAN DEBUG <000f> fsm.c:322 lchan(0-0-2-SDCCH8-4)[0x55e1591e21b0]{WAIT_RF_RELEASE_ACK}: Timeout of T3111
DCHAN DEBUG <000f> lchan_fsm.c:1574 lchan(0-0-2-SDCCH8-4)[0x55e1591e21b0]{WAIT_RF_RELEASE_ACK}: (type=SDCCH) Handling failure, will then transition to state BORKEN
DCHAN ERROR <000f> lchan_fsm.c:81 lchan(0-0-2-SDCCH8-4)[0x55e1591e21b0]{WAIT_RF_RELEASE_ACK}: (type=SDCCH) lchan failure in state WAIT_RF_RELEASE_ACK: Timeout
DCHAN DEBUG <000f> lchan_fsm.c:1574 lchan(0-0-2-SDCCH8-4)[0x55e1591e21b0]{WAIT_RF_RELEASE_ACK}: state_chg to BORKEN

I think “rach max-delay” would be better. In case of the Access Burst we’re dealing with delay (in symbol periods), not an advance yet. Regarding the reasonable default value, I think 63 would be safe to assume.

Adding a DEBUG level osmo-bsc.log, Maybe grepping on it might help. I just let it run until it was about 100M

This time it happens because of a different problem:

DCHAN DEBUG <000f> fsm.c:322 lchan(0-0-2-SDCCH8-4)[0x55e1591e21b0]{WAIT_RF_RELEASE_ACK}: Timeout of T3111
DCHAN DEBUG <000f> lchan_fsm.c:1574 lchan(0-0-2-SDCCH8-4)[0x55e1591e21b0]{WAIT_RF_RELEASE_ACK}: (type=SDCCH) Handling failure, will then transition to state BORKEN
DCHAN ERROR <000f> lchan_fsm.c:81 lchan(0-0-2-SDCCH8-4)[0x55e1591e21b0]{WAIT_RF_RELEASE_ACK}: (type=SDCCH) lchan failure in state WAIT_RF_RELEASE_ACK: Timeout
DCHAN DEBUG <000f> lchan_fsm.c:1574 lchan(0-0-2-SDCCH8-4)[0x55e1591e21b0]{WAIT_RF_RELEASE_ACK}: state_chg to BORKEN
The value of T3111 appears to be too tight for your setup? Assuming that you're using the default:

```
OsmoBSC# show timer net T3111
net: T3111 = 2 s        Wait time before RSL RF Channel Release (default: 2 s)
```

Please try again with a higher value, like 4 or even 8.

#12 - 03/28/2021 11:35 PM - fixeria

laforge wrote:

- the BTS rejects RSL CHANnel ACTIVation by sending NACK
- osmo-bsc gets confused and treats the timeslot as 'BROKEN'

osmo-bsc should for sure not get confused by a CHAN ACT NACK, is there a separate bug here?

TBH, I don't know. The current approach seems to be logical and safe from the FSM point of view. What would probably make sense is some kind of watchdog that would try to un-BORKE BORKen timeslots after a certain time. This can be done by activating a broken sub-slot and releasing it immediately. If the BTS still refuses to activate it, then it's completely BORKen and the second attempt to un-BORKE can be postponed further. What do you think?

#13 - 04/02/2021 07:37 AM - keith

Submitted https://gerrit.osmocom.org/c/osmo-bsc/+/23574

If that flies, re this ticket, I would still like to see the watchdog idea implemented.

#14 - 04/02/2021 07:38 AM - keith

- % Done changed from 0 to 50

#15 - 04/04/2021 08:21 PM - keith

- Related to Feature #5106: Watchdog that would try to un-BORKE BORKen timeslots added

#16 - 04/04/2021 08:22 PM - keith

- Due date set to 04/04/2021
- Status changed from In Progress to Resolved
- % Done changed from 50 to 100

Patch merged, resolved.

Files

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>bsc-e1.pcap</td>
<td>985 KB</td>
<td>03/26/2021</td>
<td>keith</td>
</tr>
<tr>
<td>osmo-bsc.cfg</td>
<td>5.54 KB</td>
<td>03/26/2021</td>
<td>keith</td>
</tr>
<tr>
<td>lur-bomb.pcap</td>
<td>1.37 MB</td>
<td>03/28/2021</td>
<td>keith</td>
</tr>
<tr>
<td>osmo-bsc.log.gz</td>
<td>4.21 MB</td>
<td>03/28/2021</td>
<td>keith</td>
</tr>
</tbody>
</table>