

## GDP - Bug #64

### gdplogd to gdp-router-click communications getting out of sync

10/07/2016 08:13 AM - Eric Allman

<b>Status:</b>	New	<b>Start date:</b>	10/07/2016
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Eric Allman	<b>% Done:</b>	0%
<b>Category:</b>	gdplogd	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			

#### Description

(It's not clear if this is a gdplogd or a gdp-router-click bug, but gdplogd is my first guess.)

Logs on gdp-01 seem to have hung again. In investigating the problem, I found the following log messages:

```
2016-10-06 22:57:24.546063 New connection on 7, read = 80
2016-10-06 22:57:24.546076 Received PDU fd 7 cmd 1 len 80 (sig 0, opt 0, data 0)
2016-10-06 22:57:24.546087 ClientAdvert: Received GDP_CMD_ADVERTISE A38z-NAZhwSORqoqJ0tgDqqE2gp0WZxWJo05d4njtwI fd 7
2016-10-06 22:57:24.546099 ClientAdvert (ADVERTISE A38z-NAZhwSORqoqJ0tgDqqE2gp0WZxWJo05d4njtwI): new clients 1 length 32 (I am P)
2016-10-06 22:57:24.546121 Sending NEW_CLIENT_PRIMARY (A38z-NAZhwSORqoqJ0tgDqqE2gp0WZxWJo05d4njtwI) node 128.32.62.35:8007 fd 11
2016-10-06 22:57:24.546133 Sending NEW_CLIENT_PRIMARY (A38z-NAZhwSORqoqJ0tgDqqE2gp0WZxWJo05d4njtwI) node 128.32.33.232:8007 fd 12
2016-10-06 22:57:24.546144 Sending NEW_CLIENT_PRIMARY (A38z-NAZhwSORqoqJ0tgDqqE2gp0WZxWJo05d4njtwI) node 128.32.62.174:8007 fd 13
```

This indicates a new connection from a client with id A38z-NAZhwSORqoqJ0tgDqqE2gp0WZxWJo05d4njtwI connecting on file descriptor 7. It advertises, and that advertisement is sent on to the other routers.

```
2016-10-06 22:57:24.546460 Received PDU fd 7 cmd 68 len 80 (sig 0, opt 0, data 0)
2016-10-06 22:57:24.546847 Received PDU fd 10 cmd 128 len 518 (sig 0, opt 8, data 430)
2016-10-06 22:57:24.547457 Received PDU fd 7 cmd 70 len 88 (sig 0, opt 8, data 0)
2016-10-06 22:57:24.547755 Received PDU fd 10 cmd 133 len 542 (sig 78, opt 24, data 360)
```

OPEN\_RO followed by the first of about 35 READs ...

```
2016-10-06 22:57:24.684212 Received PDU fd 7 cmd 70 len 88 (sig 0, opt 8, data 0)
2016-10-06 22:57:24.684442 Received PDU fd 10 cmd 133 len 541 (sig 78, opt 24, data 359)
2016-10-06 22:57:24.684974 Received PDU fd 7 cmd 73 len 92 (sig 0, opt 8, data 4)
2016-10-06 22:57:24.685177 Received PDU fd 10 cmd 128 len 88 (sig 0, opt 8, data 0)
2016-10-06 22:57:24.686574 Received PDU fd 10 cmd 133 len 542 (sig 78, opt 24, data 360)
```

READ, ACK, MULTIREAD, ACK, followed by over one million ACK\_CONTENT responses containing the data, then ...

```
2016-10-06 23:04:23.547832 Received PDU fd 10 cmd 133 len 539 (sig 77, opt 24, data 358)
2016-10-06 23:04:23.547846 Received PDU fd 10 cmd 130 len 181 (sig 77, opt 24, data 0)
```

the last ACK\_CONTENT and an ACK\_DELETED to indicate the end of the multiread, and then ...

```
2016-10-06 23:04:23.547857 Bogus version 0 fd 10
2016-10-06 23:04:23.547868 Bogus version -112 fd 10
2016-10-06 23:04:23.547879 Bogus version 0 fd 10
2016-10-06 23:04:23.547890 Bogus version 112 fd 10
2016-10-06 23:04:23.547901 Bogus version 120 fd 10
2016-10-06 23:04:23.547912 Received PDU fd 10 cmd 111 len 685778368 (sig 395, opt 408, data 685777
485)
2016-10-06 23:04:23.547923 Incomplete PDU fd 10 pduLen 685778368 bufferDataLen 4124
```

This represents five reads by the router that had an incorrect version number from gdplogd, indicating that the communication is out of sync. It then (probably by chance) found an octet with a version number that matched, and then read a command 111 (which is not defined) with a completely outrageous PDU length of well over 685 megabytes. At this point the damage is done, but it continues with:

```
2016-10-06 23:04:27.120866 Received PDU fd 13 cmd 3 len 1728 (sig 0, opt 0, data 1648)
2016-10-06 23:04:27.120896 Received NEW_CLIENT_PRIMARY from 128.32.62.174:8007
2016-10-06 23:04:27.120908     for zeIF_aJwtiP5xZ5bQUUS0dcYC7LfMaMComlK_5lRmCI, nc 50
2016-10-06 23:04:28.231778 Received PDU fd 12 cmd 3 len 128 (sig 0, opt 0, data 48)
2016-10-06 23:04:28.231811 Received ping, fd 12
2016-10-06 23:04:28.267123 Received PDU fd 24 cmd 70 len 80 (sig 0, opt 0, data 0)
```

This appears to be unrelated traffic from other file descriptors, at which point nearly all the traffic looks like:

```
2016-10-06 23:04:28.267665 Received PDU fd 10 cmd 111 len 685778368 (sig 395, opt 408, data 685777
485)
2016-10-06 23:04:28.267690 Incomplete PDU fd 10 pduLen 685778368 bufferDataLen 4666
2016-10-06 23:04:31.407746 Received PDU fd 10 cmd 111 len 685778368 (sig 395, opt 408, data 685777
485)
2016-10-06 23:04:31.407784 Incomplete PDU fd 10 pduLen 685778368 bufferDataLen 21050
```

Note that this means that the router is still trying to read the 685,778,368 octet PDU. This goes until I restarted gdplogd.

The only gdplogd output during that period was

```
2016-10-06 23:01:20.478845 2016-10-06 16:01:20.478775 -0700 gdplogd: post_subscribe: bad read: SEV
ERE: corrupt GCL index [Berkeley:Swarm-GDP:18]
2016-10-06 23:01:20.478894
2016-10-06 23:01:20.478908 disk_read_by_recno: recno mismatch: wanted 1632004, got 1631451_gdp_pdu
_out(ACK_DATA_DEL): siglen = 78 but only 0 available
2016-10-06 23:01:20.478921     basemd = (nil)
2016-10-06 23:01:21.204271 2016-10-06 16:01:21.204145 -0700 gdplogd: post_subscribe: bad read: SEV
ERE: corrupt GCL index [Berkeley:Swarm-GDP:18]
2016-10-06 23:01:21.204316
2016-10-06 23:01:21.204329 disk_read_by_recno: recno mismatch: wanted 1737242, got 1736677_gdp_pdu
_out(ACK_DATA_DEL): siglen = 77 but only 0 available
2016-10-06 23:01:21.204341     basemd = (nil)
2016-10-06 23:01:31.197799 gdp_pdu_proc_resp: discarding PDU for unknown GCL
2016-10-06 23:01:31.197837     994PN_UvHlPRftodcxgi_n3GHSboSXncDNFrflQojk
2016-10-06 23:01:31.407461 _gdp_advertise => OK
2016-10-06 23:01:31.407497 logd_advertise_all => OK
2016-10-06 23:01:35.637517 gdp_pdu_proc_resp: discarding PDU for unknown GCL
2016-10-06 23:01:35.637552     Ncqy30_Oe9aoV7ImNgTiekg0BGrSRExN06bKJfxRTuM
2016-10-06 23:02:01.407521 _gdp_advertise => OK
2016-10-06 23:02:01.407559 logd_advertise_all => OK
2016-10-06 23:02:23.699555 gdp_pdu_proc_resp: discarding PDU for unknown GCL
2016-10-06 23:02:23.699592     8ETZgBccEXv-Ywjg4Ej_-D0HL1OLFYhAn6L0t77OryU
2016-10-06 23:02:24.055028 gdp_pdu_proc_resp: discarding PDU for unknown GCL
2016-10-06 23:02:24.055063     a4iZhjw4xjHgviLoGh-yJQQ0v64xeS-xfzAHB24F8dc
```

```
2016-10-06 23:02:31.407999 _gdp_advertise => OK
2016-10-06 23:02:31.408050 logd_advertise_all => OK
2016-10-06 23:03:01.406446 _gdp_advertise => OK
2016-10-06 23:03:01.406495 logd_advertise_all => OK
2016-10-06 23:03:06.163754 gdp_pdu_proc_resp: discarding PDU for unknown GCL
2016-10-06 23:03:06.163789      egCC2RRiFjv4UjB4Q5rJm003vTCBrCYUB8rdMaoITc0
```

Unfortunately there is no way to tell if these correspond to this problem, but the record numbers in question are plausible.

## History

---

### #1 - 10/14/2016 01:02 PM - Eric Allman

gdplugd has been updated so that it doesn't complain if records arrive out of order. Unfortunately this also means that multiple writers are almost impossible to detect. Additional checking should make this less painful. This adds a new physical I/O entry point `recno_exists`, which, given a record number, returns whether there is already data associated with that record number. This of course does not allow "branches" in logs. See [commit:81af2bc3](#).

I doubt this fixes the problem, but may help ameliorate it.