

GDP - Support #138

_gdp_chan_advert_commit was not really committing?

11/30/2019 10:55 PM - Jason XiangJun

Status: Closed	
Priority: Normal	
Assignee: Eric Allman	
Category:	
Target version:	
Description	
gdp_chain.c	
<pre>EP_STAT _gdp_chan_advert_commit(gdp_chan_t *chan) { // eventually, flush out any pending piggybacked advertisements return EP_STAT_OK; }</pre>	
<p>The code above does not do the actual commit. It merely returns an OK status to the caller. This function is potentially confusing. For example, in the following code</p>	
<pre>EP_STAT _gdp_advertise_me(gdp_chan_t *chan, int cmd, void *adata) { ep_app_error("123123123\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$"); EP_STAT estat; gdp_chan_advert_cr_t *cr_cb = NULL; //XXX XXX gdp_adcert_t *adcert = NULL; //XXX XXX estat = _gdp_chan_advertise(chan, _GdpMyRoutingName, adcert, cr_cb, adata); if (EP_STAT_ISOK(estat)) estat = _gdp_chan_advert_commit(chan); return estat; }</pre>	
<p>The data was pushed to the socket during <code>_gdp_chan_advertise()</code>, <code>_gdp_chan_advert_commit</code> just sets the return status. Its naming is somewhat misleading.</p>	

History

#1 - 12/02/2019 12:49 PM - Eric Allman

- Status changed from New to In Progress
- Assignee set to Eric Allman

Excellent point, the name is a misnomer. I'll change it to `_gdp_chan_advert_flush`, which is more accurate. However, I'll wait until the semester is complete to avoid making life hard on students who may be trying to use the system for projects.

Just to be clear, the point of this API is to bunch multiple advertisements into a single PDU, in particular when a log server instance starts up which may publish thousands of advertisements all at once. Since AdCerts do not involve a cryptographic exchange it is easy to consolidate them into a small number of PDUs.

Even longer term, the hope is to include persistence in the routing layer such that when a log server starts up it only has to identify itself; from that the routing layer has an idea of what it had previously, so the only thing needed is to validate that there have been no other changes. This can be done by traversing a Merkel tree — if the top node matches then there have been no changes. The details, of course, are more subtle.

#2 - 01/07/2020 12:18 PM - Eric Allman

This has been changed in 2.2.2 (with a #define for back compatibility).

#3 - 01/07/2020 12:18 PM - Eric Allman

- *Status changed from In Progress to Closed*