ACNET Service - Feature #12735

Add support for large messages

05/20/2016 09:57 AM - Richard Neswold

Status:	Assigned	**Start date:**	08/28/2018
Priority:	Normal	**Due date:**	
Assignee:	Richard Neswold	**% Done:**	0%
Category:	acnetd	**Estimated time:**	0.00 hour
Target version:	acnet v2.0		

Description

Early 2015, we developed a protocol to allow ACNET to send and receive messages larger than 64K. acnetd needs to implement this feature.

Subtasks:

Feature # 20698: Large Messages: Support the client API **New**

History

#1 - 02/24/2017 04:06 PM - Richard Neswold

acnetd source has been placed in Redmine. The master branch is what's running in the control system. I created a large-packets branch which contains the efforts of this issue.

#2 - 08/27/2018 01:29 PM - Richard Neswold

- Category set to acnetd

- Target version set to acnet v2.0

This is a major change so we target it for v2.0.

#3 - 08/28/2018 01:43 PM - Richard Neswold

- Due date set to 08/28/2018

- Start date changed from 05/20/2016 to 08/28/2018

due to changes in a related task: #20698

#4 - 03/18/2020 10:10 AM - Beau Harrison

I'm adding a relevant conversation from Slack for reference:

> Beau Harrison:
> https://en.m.wikipedia.org/wiki/Stream_Control_Transmission_Protocol
>
> > Rich Neswold:
> > Happened across this post and reread it. This could be the answer for ACNET Large Message support! It's supported by Linux, BSD, OSX, Windows and VxWorks 6.4 (and above.)
> > They talk about one SCTP connection that can open "channels" so that, while one process is writing a large message, smaller messages can be written on other channels (i.e. the network stack will interleave the message chunks and rebuild them on the other side.) This is exactly what we were trying to do!
> > The downside is that all ACNET implementations would need to be tweaked/rewritten whereas our solution is backward compatible with current ACNET. However, it seems insane not to use something that's already written and a standard.

Files

ACNet UDP Long Messages.pdf	104 KB	05/20/2016	Richard Neswold