Long term monitoring and auto-saving of data.

Saved From: https://www.pingman.com/kb/article/long-term-monitoring-and-auto-saving-of-data-44.html

Question

I'd like to configure PingPlotter to run continously. Are there any guidelines on how to configure PingPlotter to auto-save data?

Solution

Note: The following information applies to versions 4.12.0 and older of PingPlotter. Version 5 introduces a completely new method for auto-saving of data, which we cover in more detail here.

Continuous Monitoring Guidelines

We have some basic guidelines, although your situation will almost certainly be different than what we do here at Pingman Tools.

Setting up your memory footprint

If you run continuously, at some point PingPlotter will exhaust your system memory unless you give it guidelines on how much memory to keep around.

- Determine how often you want to sample. 2.5 seconds gives a good amount of accuracy without too much data. Some people use 10 seconds. Anything much longer and you might miss problems.
- We find that having 4 to 5 days of data in memory at a time works well. There are 86,400 seconds in a day, 432,000 seconds in 5 days. Divide this 432,000 by your trace interval. For 2.5 seconds, this gives us 172800 samples in 5 days.
- In PingPlotter Standard, go to Edit -> Options, General tab. In PingPlotter Pro, go to Edit -> Options, then the Auto-save section of the configuration you're interested in (Default Settings, for example). Enter your calculated number in 'Maximum samples to hold in memory'. 172800 samples takes up roughly 10 to 15 megs of RAM in memory, which puts the PingPlotter memory footprint around 40 megs total (it keeps multiple copies of the data in RAM at some points, and general overhead). This is workable for just about any workstation.

Setting up PingPlotter to save data

With 4 to 5 days of data in memory, each save of data will have all of this, which puts each save file around 1 to 3 megabytes. Having one file per day gives you easy access to a day's data, along with the previous 4 days for good analysis. We suggest saving every 30 to 60 minutes, with a filename like this: c:\ppdata\$dest\$dest $date.pp2

Set filename to 'c:\ppdata\$dest\$dest $date.pp2'. If you're currently tracing to a target, floating over the filename field will show you a hint of the file that would be actually saved. Make sure you specify an absolute path on your local hard drive.

This knowledgebase article and content copyright by Pingman Tools, LLC

- In PingPlotter Standard, go to Edit -> Options, Auto-save tab. For PingPlotter Pro, go to Edit -> Options, then go to the Auto-save section of the configuration you're interested in.
- Turn on 'Auto-save data'
- Set 'Save Interval' for '30 minutes'

We set up '30 minutes' for a save interval. The filename controls how often we create a new save file. You can include $hour in the save file name to get a new file every hour. If the file already exists, it will be overwritten.

This will give you a new file each day with 5 days of data in it. Each day's file will be missing the last few minutes of the day as the 30 minute save interval may hit at 23:35 or 23:59, but that data will always be stored in the *next* day's file.

Feel free to tweak these settings however you want. This is just a discussion of some possible starting points.

Autosaving with a lot of targets

If you're using PingPlotter Pro and monitoring a lot of targets, you'll need to be cognizant of the impact of auto-saving a lot of targets. Each time the auto-save timer fires, PingPlotter completely overwrites each of your auto-save files. With a lot of data in memory and a lot of targets, this can take a few seconds. As you approach the memory limit of a machine, this can often mean memory swaps occur as well, to pull in the old samples and write them out to a file. On some machines, this can take a minute - during which time tracing can stop and the GUI can become unresponsive.

The best way to manage this is to limit the number of samples in memory - if you can keep just a couple of days in memory for each target you'll still have the history and you'll get better performance (and have more available memory on that machine).

This knowledgebase article and content copyright by Pingman Tools, LLC