PingER data flow at SLAC

Table of Contents

Introduction

This page describes, how the data collected by the SLAC PingER site, ends up as information, which is used by pingtable, motion charts, intensity maps and several other applications. Various scripts are used in turn to generate the data. The scripts in their turn are driven by a trscrontab that executes on pinger@pinger.slac.stanford.edu.

Scripts

PingER scripts are available from /afs/slac/package/pinger/.

- Note that times (and dates such as month, year in filenames) are GMT.
- TimePing:** /afs/slac/package/pinger/timeping.pl
  - Represents the old script to perform ping measurements and store raw data.
  - a caode.Stores raw data on nfs at:

```
/nfs/slac/g/net/pinger/pinger_mon_data/ping-<YYYY>-<MM>.txt
```

  - The following files were moved from /nfs/slac/g/net/pinger/pinger_mon_data to /nfs/slac/g/net/pinger/pingerdata/hep/data.unite /pinger_mon_data:

```
-rw-rw-r-- 1 pinger iepm      3271 Sep 28  2006 countries.txt
-rw-r--r-- 1 pinger iepm   5149872 Jan  3  2013 gl-2008
-rw-r--r-- 1 pinger iepm   1211063 Apr  1  2010 ping-1970-01.txt
-rw-rw-r-- 1 pinger iepm  99615499 Jan 31  2005 ping-2005-01.txt
-rw-r--r-- 1 pinger iepm 138793772 Feb 28  2005 ping-2005-02.txt
-rw-r--r-- 1 pinger iepm 158497470 Mar 31  2005 ping-2005-03.txt
...
-rw-r--r-- 1 pinger iepm 295723517 Mar 31  2010 ping-2010-03.txt
-rw-r--r-- 1 pinger iepm  32801520 Apr  4  2010 ping-2010-04.txt
-rw-rw-r-- 1 pinger iepm    126894 Sep 28  2006 pingtable.txt
```

- PingER2:
  - /afs/slac/package/pinger/pinger2/share/pinger/pinger2.pl: is the new/current tool to perform ping measurements and store raw data.
    - At SLAC pinger2.pl is called by /afs/slac.stanford.edu/package/pinger/pinger2/share/pinger/pinger_startup.pl
      This is to check that all is well (pinger2.pl exists, pinger.xml exists and pinger2.pl is not already/still running).
      It executes:

      /usr/bin/perl /afs/slac/package/pinger/pinger2/share/pinger/pinger2.pl >/afs/slac/package/pinger/pinger2/share/pinger /pingerCronStat.stdout

      Look at /afs/slac/package/pinger/pinger2/share/pinger/pingerCronStat.stdout to see how it is doing.
    - It appends the measurements to a file of the form (also see below):
      - /nfs/slac/g/net/pinger/pinger2/data/ping-<yyyy>-<mm>.txt with records looking like:

        - pinger.slac.stanford.edu 134.79.104.80 www.sonabel.bf 212.52.147.202 100 1693526406 10 10 248.170 254.193 266.091 1 2 3 4 5 6 7 8 9 10 256 250 266 255 252 257 248 252 249 253
        - One can use https://www.epochconverter.com/ to translate the epoch time (e.g. in the above 1693526406) to GMT.
    - It "use"'s

      use XML::Simple;
      use Data::Dumper;
      use Fcntl;
      use File::stat;
      use Net::Domain qw(hostname hostfqdn hostdomain);
    - Since it is run under perl, it shows up in top as:

```
   PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+
COMMAND
 15038 pinger    20   0 3769872   3.5g  11604 R  5.8  7.3 231:43.85
perl
112054 ytl       20   0  202832  10032   1372 S  0.1  0.0   8:52.46 sshd
```

- See http://confluence.slac.stanford.edu/display/IEPM/PingER2+at+SLAC]
- At SLAC stores raw data on nfs at:

```
/nfs/slac/g/net/pinger/pinger2/data/ping-<YYYY>-<MM>.txt
e.g.
/nfs/slac/g/net/pinger/pinger2/data/ping-2021-08.txt
lines aree of the form:
pinger.slac.stanford.edu 134.79.104.80 www.alief.dz 194.163.133.136 100 1627776002 10 10 148.694
149.496 152.268 1 2 3 4 5 6 7 8 9 10 148 152 148 150 148 148 151 148 148 148
```

  This file is opened and written to for each set of pings in pingers.pl via the sub log_it in pinger2.pl. Failure to access the file will result in the following message example in the CronStat.stderr file:

```
Wed Sep  8 17:09:03 2021 Can't open PINGDATA /afs/slac/package/pinger/pinger2/share/pinger/ping-
2021-09.txt: No such file or directory
```

- At SLAC:
    - Information on Pinger hosts is kept in an Oracle database called NODEDETAILS
    - The various target host files are updated from NODEDETAILS by a trscrontab job scheduled at 08:00, 13:00, 18:00, 23:00 hours calling
      /afs/slac/package/pinger/write_offsitenodes.pl > /afs/slac/g/www/www-iepm/pinger/host-warnings.txt 2>&1 and taking ~ 17 mins
    - write_offsitenodes.pl reads input from the PingER Guthrie NODEDETAILS database and writes:
      a file of nodes /afs/slac.stanford.edu/www/comp/net/mon/offsite.nodes

      a file of beacons: /afs/slac/g/www/www-iepm/pinger/beacons.txt, this is accessed via https://www-iepm.slac.stanford.edu/pinger/beacons.txt

      an XML file for worldwide (non-SLAC) pingER monitoring sites: /afs/slac.stanford.edu/g/www/www-iepm/pinger/pinger.xml, this is accessed via the URL http://www-iepm.slac.stanford.edu/pinger/pinger.xml

      an XML file for the local SLAC pinger2 monitoring: /afs/slac.stanford.edu/package/pinger/pinger2/share/pinger/pinger.xml (includes all enabled i.e. M, B, NOT-SET)

      A backup is also created at /afs/slac.stanford.edu/www/comp/net/mon/offsite.nodes.back
    - The ping commands are:

      <PingV4Cmd>/bin/ping -n -w %deadline -c %count -i %interval -s %packetsize %destination</PingV4Cmd>
      with defaults %deadline=30, %ccount 10, %packetsize 100
      <PingV6Cmd>/bin/ping6 -n -w %deadline -c %count -i %interval -s %packetsize %destination</PingV6Cmd>
    - The log file is at /afs/slac.stanford.edu/package/pinger/pinger2/share/pinger/pingerCronStat.stdout
    - pinger2.pl gets the hosts to ping and other configuration information from the file /afs/slac.stanford.edu/package/pinger/pinger2/share/pinger/pinger.xml
    - pinger2.pl records the result of each ping by updating /nfs/slac/g/net/pinger/pinger2/data/ping-<yyyy>-<mm>.txt. This is reported by pinger2.pl as eg.:
        - log_it() logged to /nfs/slac/g/net/pinger/pinger2/data//ping-2023-09.txt for www.qatarairways.qa(213.130.112.216) 1000 bytes, sent=30, rcvd=0
        - the records appear as: pinger.slac.stanford.edu 134.79.104.80 www.pgia.ac.lk 192.248.43.131 1000 1695401249 10 10 339.582 339.945 340.100 1 2 3 4 5 6 7 8 9 10 340 339 340 339 339 339 339 339 339 340
        - Use https://www.epochconverter.com/ to convert the time stamp (seen above as 1695401249) to GMT.
    - To test
      Copy the pinger2.xml file to pinger3.xml. In pinger3.xml strip most of the hosts from <BeaconList>...</BeaconList>, <HostList>...</HostList>
      Copy pinger2.pl to pinger3.pl, in pinger3.pl edit the line:
      my $xmlfile='pinger.xml'; #$cwd/pinger.xml, configuration file for pinger2.pl
      to yield

my $xmlfile='pinger3.xml'; #$cwd/pinger.xml, configuration file for pinger2.pl
In trscrontab -e comment out /afs/slac.stanford.edu/package/pinger/pinger2/share/pinger/pinger_startup.pl and wait for next half-hour interval (by which time the current pinger2.pl should have stopped running).
Then run cd /afs/slac.stanford.edu/package/pinger/pinger2/share/pinger ; perl -d /afs/slac.stanford.edu/package/pinger/pinger2/share/pinger/pinger3.pl;

- Ping_Data
  - /afs/slac/g/www/cgi-bin/net/offsite_mon/ping_data/ping_data.pl: is called from getdata.pl
  - Represents the web interface(cgi script) used to fetch data from remote monitoring node.
  - zips the data before transfer.
- getdata_all.pl [-h monitoring_hostname] [-d days_ago] [-s start_date] [-e end_date]
  - getdata_all.pl -h pinger.slac.stanford.edu -d 5
  - getdata_all.pl -h all -s 2023-08-27 -e 2023-08-30
  - Calls getdata.pl
- getdata.pl
  - /afs/slac/package/pinger/getdata.pl
  - Typically called from /afs/slac/package/pinger/getdata_all.pl
    - Typical call getdata.pl pinger.slac.stanford.edu 2023-09-03 1 #Fetch data for 2023-09-03 usually written early morning 2023-09-04
  - Typical debug call:
    - perl -d ~cottrell/bin/getdata.pl pinger.slac.stanford.edu 2023-09-08 2
  - queries ping_data.pl http://slac.stanford.edu/cgi-bin/ping_data.pl to fetch data (e.g. from /nfs/slac/g/net/pinger/pinger2/data)
  - uses wget calls of the form:
    - /usr/bin/wget --no-check-certificate --no-verbose --tries=1 --quiet -O - 'https://www.slac.stanford.edu/cgi-bin/ping_data.pl?in_form=1&begin_hour=0&begin_min=00&begin_sec=00&begin_day=10&begin_month=09&begin_year=2023&begin_offset=&begin_point=y&end_hour=7&end_min=59&end_:wsec=59&end_day=04&end_month=06&end_year=2011&end_offset=&end_point=y'
  - Writes files of the form:
    /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-09-03.txt.gz
    - If the file length is 40 then there is a problem, it may appear as something like:

      [cottrell@cent7a ~]$ ls -l /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-09-03.txt.gz

      -rw-rw-r--. 1 pinger iepm 40 Sep 4 01:32 /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-09-03.txt.gz #Failed
      If the length is 40 then will need to run getdata.pl again to really get the data. Then run

      The response for a good file is
      [cottrell@cent7a ~]$ ls -l /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-09-02.txt.gz

      -rw-rw-r--. 1 pinger iepm 3177181 Sep 3 01:32 /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-09-02.txt.gz #Worked
    - If the file length is 40 then will need to run getdata.pl again to really get the data.
  - Since 2004 it stores old files as zipped File at

```
/nfs/slac/g/net/pinger/pingerdata/hep/data/<host>/ping-<YYYY>-<MM>-<DD>.txt.gz
e.g.
/nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2022-11-23.txt.gz
/nfs/slac/g/net/pinger/pingerdata/hep/data/pcgiga.cern.ch/ping-2006-09-28.txt.gz
/nfs/slac/g/net/pinger/pingerdata/hep/data/2001:da8:270:2018:f816:3eff:fef3:bd3/ping-2018-09-09.txt.gz
or
[pinger@pinger ~]$ ls -l /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2022-11-23.txt.gz
-rw-rw-r--. 1 pinger iepm 2720699 Nov 24 09:05 /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2022-11-23.txt.gz
```

    At the Measurement Agents (MAs) it is typically stored at: /usr/local/share/pinger/data
  - There is also a wrapper to get data from multiple days. It is at /afs/slac/package/pinger/getdata_all.pl
  - See **Restoring PingER data** for more details.
  - Data from 1997-2003 can be found in /nfs/slac/g/net/pinger/pingerdata/. The files are zipped and compressed and contain the gathered data for 100 and 1000 byte pings for each day for all monitors.

```
157cottrell@pinger:~$ls -l /nfs/slac/g/net/pinger/pingerdata/
total 11
drwxrwsr-x  2   6995 iepm  512 Dec  2  2011 1997/
drwxrwsr-x  2   6995 iepm 1024 Jan 25  2005 1998/
drwxrwsr-x  2   6995 iepm 1024 Jan 25  2005 1999/
drwxrwsr-x  2   6995 iepm 1024 Aug  3  2007 2000/
drwxrwsr-x  2   6995 iepm  512 Jan 25  2005 2001/
drwxrwsr-x  2   6995 iepm  512 Jan 25  2005 2002/
drwxrwsr-x 38   6995 iepm 1536 Feb  8  2007 2003/
drwxrwsr-x  6 iepm   iepm  512 Jul 20 15:19 hep/
drwxrwsr-x  3 pinger iepm  512 Mar  8  2012 new/
drwxrwsr-x  4   6995 iepm  512 Jan 25  2005 oldftp/
158cottrell@pinger:~$ls -l /nfs/slac/g/net/pinger/pingerdata/2002/
total 3844288
-rw-r--r-- 1 6995 iepm 349112320 Jan 25  2005 data-2002-01.tar
-rw-r--r-- 1 6995 iepm 316467200 Jan 25  2005 data-2002-02.tar
-rw-r--r-- 1 6995 iepm 332656640 Jan 25  2005 data-2002-03.tar
-rw-r--r-- 1 6995 iepm 326103040 Jan 25  2005 data-2002-04.tar
-rw-r--r-- 1 6995 iepm 347064320 Jan 25  2005 data-2002-05.tar
-rw-r--r-- 1 6995 iepm 324648960 Jan 25  2005 data-2002-06.tar
-rw-r--r-- 1 6995 iepm 319150080 Jan 25  2005 data-2002-07.tar
-rw-r--r-- 1 6995 iepm 320245760 Jan 25  2005 data-2002-08.tar
-rw-r--r-- 1 6995 iepm 336117760 Jan 25  2005 data-2002-09.tar
-rw-r--r-- 1 6995 iepm 335669760 Jan 25  2005 data-2002-10.tar
-rw-r--r-- 1 6995 iepm 303858176 Jan 25  2005 data-2002-11.tar
-rw-r--r-- 1 6995 iepm 323323904 Jan 25  2005 data-2002-12.tar
168cottrell@pinger:~$cp /nfs/slac/g/net/pinger/pingerdata/2002/data-2002-01.tar /nfs/slac/g/net
/pinger/pingerdata/hep/data.unite/
$cd /nfs/slac/g/net/pinger/pingerdata/hep/data.unite/
$tar -xvf /nfs/slac/g/net/pinger/pingerdata/hep/data.unite/data-2002-01.tar
cache01.ansp.br/ping-2002-01-01.txt.gz
cache01.ansp.br/ping-2002-01-02.txt.gz
...
yumj2.kek.jp/ping-2002-01-30.txt.gz
yumj2.kek.jp/ping-2002-01-31.txt.gz
182cottrell@pinger:/nfs/slac/g/net/pinger/pingerdata/hep/data.unite$ls
172.23.52.7/                 monitor.seecs.edu.pk/               pinger.cdacmumbai.in/
pingerlhr-pu.pern.edu.pk/
aup.seecs.edu.pk/            moore.ece.rice.edu/                 pinger.cemb.edu.pk/
pingerpwr.pern.edu.pk/
...
177cottrell@pinger:/nfs/slac/g/net/pinger/pingerdata/hep/data.unite$cp yumj2.kek.jp/ping-2002-01-
31.txt.gz /tmp/
178cottrell@pinger:/nfs/slac/g/net/pinger/pingerdata/hep/data.unite$gunzip /tmp/ping-2002-01-31.
txt.gz
180cottrell@pinger:/nfs/slac/g/net/pinger/pingerdata/hep/data.unite$tail /tmp/ping-2002-01-31.txt
yumj2.kek.jp 130.87.34.37 ultra.edu.uy 164.73.128.70 100 1012520610 10 10 379.815 381.366 384.206
0 1 2 3 4 5 6 7 8 9 380.758 380.938 382.205 380.480 380.878 384.206 379.815 381.087 382.547 380.743
yumj2.kek.jp 130.87.34.37 frcu.eun.eg 193.227.1.1 100 1012520610 10 10 364.870 399.626 446.021 0 1
2 3 4 5 6 7 8 9 398.877 395.008 388.069 396.147 416.314 392.862 398.349 399.744 364.870 446.021
```

Note that after the above, the raw input data to wrap-analyze-hourly.pl (see below) for 1998..2003 comes from /nfs/slac/g/net/pinger/pingerdata/hep/data.unite/ rather than /nfs/slac/g/net/pinger/pingerdata/hep/data/.

- There is a script /afs/slac/package/pinger/pre2004.pl that will take the data in /nfs/slac/g/net/pinger/pingerdata/<1998..2003> copy it, unzip, and untar into /nfs/slac/g/net/pinger/pingerdata/hep/data.unite/.
- There is a second script /afs/slac/package/pinger/pre2004-hourly.pl that using /afs/slac/package/pinger/analysis/analyze-all.pl reads the raw data from /afs/slac/g/net/pinger/pingerdata/hep/data.unite/ and analyzes and stores the hourly data for the selected years.

- Unite-monthly
  - There is a script /afs/slac/package/pinger/unite-monthly.pl that unites all the raw data for a month into a single file of the form /nfs/slac/g/net/pinger/pingerdata/hep/data/<host>/ping-<YYYY>-<MM>.txt. It is run from a trscronjob at 4:00am each morning.
  - The files appear as:

```
[pinger@cent7a ~]$ ls -l /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-06-03.txt.gz
-rw-rw-r--. 1 pinger iepm 3216199 Jun  4 12:47 /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-06-03.txt.gz
```

- The output data appears as:

```
cottrell@pinger $ head /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2020-03.txt
pinger.slac.stanford.edu 134.79.104.80 www.alief.dz 91.121.45.149 100 1583020804 10 10 149.672 149.979 150.649 1 2 3 4 5 6 7 8 9 10 149 149 150 150 149 149 150 149 149 149
pinger.slac.stanford.edu 134.79.104.80 www.ons.dz 196.20.66.100 100 1583020805 10 10 193.218 193.371 193.576 1 2 3 4 5 6 7 8 9 10 193 193 193 193 193 193 193 193 193 193
[pinger@pinger ~]$ tail /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2022-11.txt
pinger.slac.stanford.edu 134.79.104.80 mail.neda.af 117.55.192.5 1000 1669247047 11 10 310.965 311.160 311.281 1 2 3 4 5 6 8 9 10 11 311 311 311 311 311 311 311 310 311 311
...
Where for example you can use https://www.epochconverter.com/ to find that the epoch time 1669247047 was equivalent to
GMT: Wednesday, November 23, 2022 11:44:07 PM

The timestamp on the file appears as:
[pinger@cent7a ~]$ ls -l /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-06.txt
-rw-rw-r--. 1 pinger iepm 68740320 Jun  4 04:00 /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-06.txt
where the date is todays date at 4:00am
```

- Data formats for timeping.pl, pinger2.pl, ping_data.pl, getdata.pl and getdata_all.pl
- Check the data has been gathered:
  - /afs/slac/package/pinger/checkdata_gif.pl calls /afs/slac/package/pinger/checkdata.pl to see if there is gathered data, and then writes out a summary web page to http://www-iepm.slac.stanford.edu/monitoring/checkdata/
  - You can manually run /afs/slac/package/pinger/checkdata.pl for a particular date with debugging
    - [pinger@cent7a ~]$ perl -d ~cottrell/bin/checkdata.pl 2023-06-03 2 #Check whether data for 2023-06-03 is present with debug set to 2.
- Analyze
  - There is a group of analysis scripts, that pick up zipped data; do their analysis, aggregate data and prepare web reports.
    - Note that to be included in the analyzed data the host must be in NODEDETAILS.
  - The first script to be executed is wrap-analyze-hourly.pl (which is executed by calling the wrapper analyze-all.pl --date 1days from the trscronjob) which takes as input data the output of getdata.pl (raw data) and from this aggregates the data to by day and writes the latest to the /nfs/slac/g/net/pinger/pingerreports/hep/<metric>/ directory with the file name <metric>~~<size>~~<by>~~<yyyy>~~<mm>-<dd>.txt.X., this is referred to as the hourly analyzed data. The wrap-analyze-hourly.pl script is run daily from the trscrontab on pinger and by default analyzes the data gathered for yesterday.
  - Example of analyze-all.pl

```
[pinger@pinger ~]$ ~cottrell/bin/analyze-all.pl  --date 2023-08-01 1 #analyze the
Sat Aug 12 11:44:42 2023(0s so far) executing /afs/slac/package/pinger/analysis/wrap-analyze-
hourly.pl  --basedir /nfs/slac/g/net/pinger --usemetric --dataset hep --debug 0  --datadir /nfs
/slac/g/net/pinger/pingerdata/hep/data --size 100 --by by-node --date 2023-08-01  --set_metric 4
...
Sat Aug 12 11:44:43 2023: wrap-analyze-hourly.pl unzipped(debug=0, size=100, by=by-node, 1s so
far) (85/229)16532 lines from /nfs/slac/g/net/pinger/pingerdata/hep/data/ping.desy.de/ping-2023-08-
01.txt.gz
Read 16532 lines for $monitoring_site ping.desy.de: found 8218 valid lines,  0 corrupted lines &
140 alpha>1.1 lines
... for each metric
wrap-analyze-hourly.pl opening metric=maximum_rtt (16/1) for write to /nfs/slac/g/net/pinger
/pingerreports/hep/maximum_rtt/maximum_rtt-100-by-node-2023-08-01.txt
Wrote 1955 lines to /nfs/slac/g/net/pinger/pingerreports/hep/maximum_rtt/maximum_rtt-100-by-node-
2023-08-01.txt.gz
... new value of --by (--by node, --by site) also will be --size 100, and --size 1000
Sat Aug 12 11:49:33 2023: wrap-analyze-hourly.pl unzipped(debug=0, size=100, by=by-site, 0s so
far) (85/229)16532 lines from /nfs/slac/g/net/pinger/pingerdata/hep/data/ping.desy.de/ping-2023-08-
01.txt.gz
... Both -by site and --size 1000
wrap-analyze-hourly.pl opening metric=minimum_rtt (16/2) for write to /nfs/slac/g/net/pinger
/pingerreports/hep/minimum_rtt/minimum_rtt-1000-by-site-2023-08-05.txt
Wrote 851 lines to /nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt/minimum_rtt-1000-by-site-
2023-08-05.txt.gz
```

The file names appear as:

```
/nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt/minimum_rtt-100-by-node-2006-09-28.txt.gz
```

You can test whether the latest data is there (e.g. say today is 6/4/2023)

```
[cottrell@pinger ~]$ date
Sun Jun  4 16:32:37 PDT 2023
[cottrell@pinger ~]$ ls -l /nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt/minimum_rtt-100-by-
node-2023-06-*.txt.gz
-rw-rw-r--. 1 pinger iepm 153081 Jun  2 02:58 /nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt
/minimum_rtt-100-by-node-2023-06-01.txt.gz
-rw-rw-r--. 1 pinger iepm 154116 Jun  3 02:58 /nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt
/minimum_rtt-100-by-node-2023-06-02.txt.gz
i.e. the 2003-06-03 data is missing.
```

The output appears as:

```
cottrell@pinger $ cp /nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt/minimum_rtt-100-by-node-
2006-09-28.txt.gz /tmp/
cottrell@pinger $ zcat /tmp/minimum_rtt-100-by-node-2006-09-28.txt.gz
cottrell@pinger $ head /tmp/minimum_rtt-100-by-node-2006-09-28.txt
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
pinger.slac.stanford.edu manila.global.net.pg 547.877 548.021 547.934 548.611 548.680 548.069
549.023 548.124 548.687 547.573 548.639 549.181 547.998 548.302 547.858 547.464 548.470 548.251
547.849 547.708 548.463 547.473 547.651 547.732 pinger.slac.stanford.edu manila.global.net.pg
pinger.slac.stanford.edu lobelia.physics.wisc.edu 61.089 60.958 61.037 61.332 60.958 61.262 61.073
61.093 61.040 61.086 60.974 60.979 60.956 61.070 60.942 61.035 61.994 61.022 60.927 60.957 61.091
61.009 61.024 60.974 pinger.slac.stanford.edu lobelia.physics.wisc.edu
```

By default the above file is created once thus the directory appears as:

```
57cottrell@pinger:~>ls -l /nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt/minimum_rtt-100-by-
node-2011-05*
-rw-rw-r--  1 pinger   iepm 492144 May  2 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-01.txt.gz
-rw-rw-r--  1 pinger   iepm 545968 May  3 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-02.txt.gz
-rw-rw-r--  1 pinger   iepm 561661 May  4 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-03.txt.gz
-rw-rw-r--  1 pinger   iepm 566550 May  5 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-04.txt.gz
-rw-rw-r--  1 pinger   iepm 537127 May  6 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-05.txt.gz
-rw-rw-r--  1 pinger   iepm 538830 May  7 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-06.txt.gz
-rw-rw-r--  1 pinger   iepm 488360 May  8 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-07.txt.gz
-rw-rw-r--  1 pinger   iepm 499020 May  9 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-08.txt.gz
-rw-rw-r--  1 pinger   iepm 563840 May 10 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-09.txt.gz
-rw-rw-r--  1 pinger   iepm 583454 May 11 02:17 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-10.txt.gz
-rw-rw-r--  1 cottrell iepm 577949 May 12 22:08 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-11.txt.gz
-rw-rw-r--  1 cottrell iepm    102 May 12 17:25 /nfs/slac/g/net/pinger/pingerreports/hep
/minimum_rtt/minimum_rtt-100-by-node-2011-05-12.txt.gz
```

Example output format. Following the 1st line in the file there is 1 line like the following per day/per host pair. Between the initial and final src_name and tgt_name tokens there are 24 tokens one for each hour of the day, missing data is identified by a dot followed by a space (.), e.g.:

```
icfamon.dl.ac.uk lns62.lns.cornell.edu 108.871 . . . . . . . . 108.892 . . . . . . . . . . . .
109.620 icfamon.dl.ac.uk lns62.lns.cornell.edu
```

The first line in the file contains a label for each of the time slots (e.g. hours):

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
```

Example output filename for the minimum_rtt metric:

- The remaining analyze scripts (wrap-analyze-daily.pl, wrap-analyze-monthly.pl, wrap-analyze-allmonths.pl, and wrap-analyze-allyears.pl) take as input the data from wrap-analyze-hourly.pl, wrap-analyze-daily.pl, and wrap-analyze-allmonths.pl and create files of the form:

```
/nfs/slac/g/net/pinger/pingerreports/hep/<metric>-<size>-by-<site|node>(-<YYYY>?)(-<mm>?)(-<dd>?).
txt.gz
/nfs/slac/g/net/pinger/pingerreports/hep/<metric>-<size>-by-<site|node>-<60|120|365>days.txt.gz
/nfs/slac/g/net/pinger/pingerreports/hep/<metric>-<size>-by-<site|node>-<allmonths|allyears>.txt.gz
```

- There are ~ 16 metrics:

<option value="MOS">Mean Opinion Score</option>
<option value="alpha">Directivity</option>
<option value="average_rtt" selected>Average Round Trip Time</option>
<option value="conditional_loss_probability">Conditional Loss Probability</option>
<option value="duplicate_packets">Duplicate Packets</option>
<option value="ipdv">Inter-Packet Delay Variation</option>
<option value="iqr">Inter-Quartile Range</option>
<option value="maximum_rtt">Maximum Round Trip Time</option>

<option value="minimum_packet_loss">Minimum Packet Loss</option>
<option value="minimum_rtt">Minimum Round Trip Time</option>
<option value="out_of_order_packets">Out of Order Packets</option>
<option value="packet_loss">Packet Loss</option>
<option value="throughput">TCP Throughput (kbits/s)</option>
<option value="unpredictability">Ping Unpredictability</option>
<option value="unreachability">Ping Unreachability</option>
<option value="zero_packet_loss_frequency">Zero Packet Loss Frequency</option>

Information on these can be found at http://www.slac.stanford.edu/comp/net/wan-mon/tutorial.html.

- More information can be found on running getdata.pl and the analyze scripts to recover missing data.

- PRM
  - /afs/slac/package/pinger/prm/prm.pl
  - The script prm.pl, is used to create csv files for the analyzed data which can then be used by PingER motion charts or PingER Executive Plots.
  - The script requires a configuration file, which contains entries for all the reports prm should create. These entries are of the form:

```
<metric name>    <monitoring site>    <country||continent>    <tick>    (<filter>?)
```

  - It gets its data from pingtable.pl either by calling it locally or by accessing it via the web.
  - The reports created by prm are available online.
  - The reports are used by the metric motion plots, the metric maps and the executive plots.
  - An example of the output is seen below:

allyearly,?,1998,1999,2000,2001,2002,2003,2004,2005,2006,2007,2008,2009,2010

EDU.SLAC.STANFORD.N3-to-Afghanistan,.,.,.,.,.,.,.,.,.,.,.,12.504,4.481,9.257,5.284

EDU.SLAC.STANFORD.N3-to-Albania,.,.,.,.,.,.,.,.,.,.,.,4.949,7.133,4.590

later it is atomically copied to files in the permanent directory

A period(.) means there was no data.

There is a directory with lots of these analyses, see https://www-iepm.slac.stanford.edu/pinger/prmout/

Data

Data flow

1. The raw data is measured by pinger2.pl and gathered from the MAs using getdata_all.pl. The data is saved in /nfs/slac/g/net/pinger/pingerdata /hep/data/<MA>/ping-<YYYY>-MM-DD.txt.gz
   An example record is: pinger.slac.stanford.edu 134.79.240.30 ping.slac.stanford.edu 134.79.18.21 100 1152489601 10 10 0.255 0.341 0.467 0 1 2 3 4 5 6 7 8 9 0.287 0.380 0.467 0.391 0.327 0.387 0.291 0.332 0.255 0.299
   To unzip the file:

   [pinger@pinger ~]$ cp /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2023-09-10.txt.gz ping-2023-09-10.txt.gz

   [pinger@pinger ~]$ gunzip ping-2023-09-10.txt.gz

   [pinger@pinger ~]$ ls -l ping-2023-09-10.txt

   -rw-r--r--. 1 pinger sf 16911992 Sep 12 12:19 ping-2023-09-10.txt

   To look at the last record use:

   [pinger@pinger ~] tail ping-2023-09-10.txt

   To see when logged look at the unix GMT time, eg the 1694389609 in:

   pinger.slac.stanford.edu 134.79.104.80 www.mmbbank.com.mm 43.242.135.186 1000 1694389609 31 0

   use https://www.unixtimestamp.com/

2. The hourly data is created from the raw data in /nfs/slac/g/net/pinger/pingerdata/hep/data/<site> (e.g. /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-2021-03-06.txt.gz) by /afs/slac/package/pinger/analysis/analyze-all.pl and saved in the /nfs/slac/g/net/pinger /pingerreports/hep/<metric>/ directory with rhe file name <metric>-<size>-<by>-<yyyy>-<mm>-<dd>.txt.gz. E.g. output filename: /nfs/slac/g/net /pinger/pingerreports/hep/minimum_rtt/minimum_rtt-100-by-node-2006-09-28.txt.gz. There are 24 hour lines per day/per host pair that appear as e.g.: icfamon.dl.ac.uk lns62.lns.cornell.edu 108.871 ... 109.620 icfamon.dl.ac.uk lns62.lns.cornell.edu
3. The daily data is created from the hourly data by /afs/slac/package/pinger/analysis/wrap-analyze-daily.pl. It creates files for 30, 120, 180 and 365 days, that are saved in files of the form: /nfs/slac/g/net/pinger/pingerreports/hep/data/[metric]/[metric]-[size]-by-[site|node]-YYYY-MM.txt.gz, e. g. /nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt/minimum_rtt-100-by-node-2007-05.txt.gz /nfs/slac/g/net/pinger/pingerreports/hep/throughput/throughput-100-by-node-120days.txt.gz
4. The monthly data is created from the daily data by /afs/slac/package/pinger/analysis/wrap-analyze-monthly.pl and the output goes to the /nfs/slac /g/net/pinger/pingerreports/hep/<metric>/ directory with the filename <metric>-<size>-<by>.txt.gz. e.g. /nfs/slac/g/net/pinger/pingerreports/hep /packet_loss/packet_loss-100-by-node.txt.gz
5. The yearly data is created from the monthly data by /afs/slac/package/pinger/analysis/wrap-analyze-allyears.pl and it creates files such as /nfs /slac/g/net/pinger/pingerreports/hep/average_rtt/average_rtt-100-by-node-allyears.txt.gz. The output file lines appear as:

```
1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018
2019 2020 2021
121.52.146.180 adl-a-ext1.aarnet.net.au . . . . . . . . . . . . . . . . 314.0620 338.4232 281.2957 . .
. . 121.52.146.180 adl-a-ext1.aarnet.net.au
121.52.146.180 airuniversity.seecs.edu.pk . . . . . . . . . . . . . . . . . 5.8450 . . . . .
121.52.146.180 airuniversity.seecs.edu.pk
...
```

```
/nfs/slac/g/net/pinger/pinger_mon_data/ping-<YYYY>-<MM>.txt contains raw data from 2005-2010, just for SLAC.
/nfs/slac/g/net/pinger/pinger2/data/ping-<YYYY>-<MM>.txt contains raw data from 2009-present, just for SLAC
/nfs/slac/g/net/pinger/pingerdata/<data> contains data from 1997-2003 contains raw data
/nfs/slac/g/net/pinger/pingerdata/hep/data/<host>/ contains raw data 1997-200
```

Data volumes Sep 2014

The total PingER data volume is about 550Gbytes.

We estimate that there are about 60 GBytes of uncompressed hourly data for 100 Byte pings by node, as of September 2014. One would estimate to about quadruple that if one added 1000 byte pings and by site. See Volume of PingER data Sep 2014.

Archiving

See Archiving PingER data by tar for retrieval by anonymous ftp

Missing data

Anomalies

We have spotted anomalies between the values reported by:

- table.pl and pingtable.pl
- and pingtable.pl at SLAC and pingtable.pl at SEECS

They are discussed and explained in the Anomalies report.

Problems

I get the error message below:

```
Your "cron" job


/afs/slac/package/pinger/analysis/wrap-analyze-daily.pl
--basedir /nfs/slac/g/net/pinger --usemetric --dataset hep --by by-site --size 1000

produced the following output:

Thu Dec  3 05:00:02 2015 Warning /nfs/slac/g/net/pinger/pingerreports/hep/minimum_rtt/minimum_rtt-1000-by-site-
2015-12-01.txt.gz does not exist
```

Since the trscronjob analyze-all.pl --date 1days only reads and analyzes the most recent day's raw data if the job fails to run, then next days there will be missing data and you wilr get the above message. To recover the missing daily data run analyze-all.pl --date 2015-12-01 for the missing date (in this example 2015-12-01).

Backup

On 3/9/2012, we requested unix-admin@slac.stanford.edu to backup /nfs/slac/g/net/pinger/ on a regular basis. This was added to the nightly backup by Andrew May.

Order of running jobs in trscrontab

```
##############Basic PingER files scripts follow ######################
#Update nodes.cf (Perl dump of Guthrie config) from the NODEDETAIL Guthrie database, add by Cottrell 1/31/07
Though node.pl is called from runall.pl, runall.pl is not called until 5am, so it is updated here first to be
sure.It takes about 3 seconds to run
pinger 30 0,6,12,18 * * * /afs/slac/package/pinger/guthrie/node.pl -o > /afs/slac/package/pinger/nodes.cf
#Takes about 30 secs
# Create the offsite.nodes list of nodes to ping from Guthrie NODEDETAILS database, add by Cottrell 1/3/2107
takes ~25 minutes
# The following files are created:
# /afs/slac.stanford.edu/www/comp/net/mon/offsite.nodes #txt file of all node, format: name address #comment
# /afs/slac.stanford.edu/g/www/www-iepm/pinger/beacons.txt #txt file of Beacons
# /afs/slac.stanford.edu/g/www/www-iepm/pinger/pinger.xml #xml file for non SLAC MA
# /afs/slac.stanford.edu/package/pinger/pinger2/share/pinger/pinger.xml  #xml file for SLAC MA
lnxcron 00 23 * * * /afs/slac/package/pinger/write_offsitenodes.pl > /afs/slac/g/www/www-iepm/pinger/host-
warnings.txt 2>&1 #takes ~15  minutes on Pinger
#--------------Normal complete set for getdata.pl----------------------
pinger;200 32 1 * * * /afs/slac/package/pinger/getdata.pl > /afs/slac/g/www/www-iepm/pinger/slaconly/getdata.
err #Takes ~20 minutes 7/31/2019
#Minimal subset is /afs/slac/package/pinger/getdata_all.pl -h pinger.slac.stanford.edu #Takes < 1 minute for
pinger.slac.stanford.edu
#-----------------analyze-hourly.pl runs first for both 100B & 1000B pings, by-host, by
site-------------------------
lnxcron;120 55 2 * * * Time-year.2023c
#Takes 25:14 10/20/2011 (55 mins 9/21/2011, 70 minutes 5/11/2018)
#-----------------analyze-daily-----------------------------------------
pinger;300 30 4 * * * /afs/slac/package/pinger/analysis/wrap-analyze-daily.pl --basedir /nfs/slac/g/net/pinger
--use/metric --dataset hep # Takes ~ 5 hours for whole month on pinger 11/10/2012
#-----------------analyze-monthly.pl for defaults by-node, size=100B takes about 60 minutes-------------
lnxcron 12 10 * * * /usr/local/bin/bsub -W 200 -o /dev/null -q xxl /afs/slac/package/pinger/analysis/wrap-
analyze-monthly.pl --basedir /nfs/slac/g/net/pinger --usemetric --dataset hep > /dev/null #Takes about 55mins 9
/19/2011, 61 mins 10/9/2019 on pinger or 7 mins on rhel6 11/13/2011.
#------------------sites-per-country-------------------------------------
pinger 00 09 * * * /afs/slac/package/pinger/sites-per-country.pl #Takes anout 10 minutes, run after analyze
monthly
```

There is a script /u/sf/cottrell/bin/manual-upd8-sites-per-country.pl that will run in sequence the minimal set of scripts needed to update sites-per-country.html.

Checks:

1. View pinger.xml at SLAC
2. View log of measurements from SLAC, or view form
3. View gathered data for SLAC at /nfs/slac/g/net/pinger/pingerdata/hep/data/pinger.slac.stanford.edu/ping-<YYYY>-<MM>-<DD>.txt.gz
4. View analyzed hourly data
5. View analyzed daily data
6. View analyzed monthly data
7. View sites per country

Accessing PinGER data

Overview

The simple way to get at the data is not to use the archive (since the data needs cleansing) but to use the form https://www.slac.stanford.edu/cgi-bin/pingtable.pl . This enables you to select the metric, packet size, source (Measurement Agent - MA), the target group, the time aggregation, etc. There is also a filter which can be used for removing bad data from the monthly and hence yearly data. Bad data is typically from a host that at first looks to be in a certain region (e.g. China) but the RTT as seen from SLAC is impossibly small.

Once you have the relevant table then cut and paste it into Excel and use that for the manipulation.
There is also a script prm.pl that uses the pingtable.pl form (via wget). This is used to access pingtable.pl multiple times to aggregate the data from an MA to multiple target groups for each metric, by year or by month.

- The script prm.pl, is used to create csv files for the analyzed data which can then be used by PingER motion charts or PingER Executive Plots.
- The script requires a configuration file, which contains entries for all the reports prm should create. These entries are of the form:

```
<metric name>    <monitoring site>    <country||continent>    <tick>    (<filter>?)
```

- It gets its data from pingtable.pl either by calling it locally or by accessing it via the web.
- The reports created by prm are available online.
- The reports are used by the metric motion plots, the metric maps and the executive plots.

An example oif the output is below:

allyearly,?,1998,1999,2000,2001,2002,2003,2004,2005,2006,2007,2008,2009,2010

EDU.SLAC.STANFORD.N3-to-Afghanistan,.,.,.,.,.,.,.,.,.,.,.,.,.,12.504,4.481,9.257,5.284

EDU.SLAC.STANFORD.N3-to-Albania,.,.,.,.,.,.,.,.,.,.,.,.,.,4.949,7.133,4.590

later it is atomically copied to files in the permanent directory

How to retrieve data from the SLAC anonymous FTP archive:

See https://confluence.slac.stanford.edu/display/IEPM/Retrieving+Archived+PingER+hourly+data+from+Anonymous+FTP
Should you decide to go ahead I am happy to help/hold your hand and improve the documentation.

Data Flow

The mechanism of gathering, archiving the data, cronjobs etc is described in https://confluence.slac.stanford.edu/display/IEPM/PingER+data+flow+at+SLAC . this also describes the raw and analyzed data.
A period(.) means there was no data.

/afs/slac/g/www/www-iepm/pinger/prmout

There is a directory with lots of these analyses, see https://www-iepm.slac.stanford.edu/pinger/prmout/

Directories

See here

/usr/local/share/pinger/ is accessed by:

- ping_data.pl - the script is called from getadata.pl as the remote MA's web server interface to get data from a remote MA. The use of /usr/local/share/pinger is only used by non SLAC MAs. It is not needed in the pinger.slac.stanford.edu MA.