

PingER Monitor node format

There are two Raw Data Formats for the data produced by the monitor,

The early one (no longer created but still in the archive):

The items are space separated:

Monitor_Host_Name	Monitor_Addr	Remote_Name	Remote_Addr	Bytes	Time	Xmt	Rcv	Min	Avg	Max
minos.slac.stanford.edu	134.79.196.100	www.lbl.gov	128.3.7.14	100	870393602	10	10	6	18	125
minos.slac.stanford.edu	134.79.196.100	ns1.cebaf.gov	129.57.32.100	100	870393602	10	10	74	93	125

Where the columns are separated by a single space.

The Bytes can be 100 or 1000 (sometimes it will be less than 100, in which case treat as 100), it is the number of bytes in each ping packet.

Time is the Unix Epochal time and is GMT (UDT).

Xmt is the number of ping packets sent.

Rcv is the number of ping packets received.

Min is the minimum response time for the packets sent (in milliseconds).

Avg is the average response time for the packets sent.

Max is the maximum response time for the packets sent.

There should always be > 7 tokens in the line. If no response to the pings were received then there will only be 8 tokens in the line and Rcv (the 8th token) will be 0.

Early versions of the data collection program lost the Remote_Addr when the remote site was not reachable. In this case the data only contains 6 tokens. It should either be ignored or treated as remote site unreachable.

The more recent (since 2008) one looks like:

Src_name	Src_addr	Dst_name	Dst_addr	Len	Unix_GMTs	sent	rcv														
min	avg	max	seqs	RTTs																	
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													

The first 11 fields are the same as before. Then for each ping response received the Sequence number is recorded, followed by the RTT for each ping

Note that though one normally asks for 10 pings to be sent since it uses the default ping on Linux the deadline feature of Linux pings means up to 30 pings can be sent while it awaits to hear 10 responses.

Also funny things happen when the pinged host returns duplicated ping responses. This needs to be documented.

Occasionally the data is mangled so it is a good idea to validate each line. A typical piece of perl to do this is:

```
my (@field)=split /\s+/, $line; #line contains a line of ping data.
#We choose to look at the validity of something that should be in the line as far down the line as possible.
#A typical line appears as:
#monitor.niit.edu.pk 203.99.50.204 www.carnet.hr 161.53.160.25 100 1171756802 10 10 223.323 224.978 226.805 1 2 3 4 5 6 7 8 9 10 226 223 226
223 223 226 223 224 226 225
next unless(defined($field[6]) && $field[6] =~ /\d/); # $field[6]=packets sent, skip line if not right
#Also validate the rough correctness of the IP address
if(!defined($field[1]) || $field[1] !~ /\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}/ ) {next;}
if(!($field[4] =~ /\d+$/)) {next;} #Verify number of bytes is a number, else skip line
#Kludge to allow default 56Byte pings to be treated as 100Byte pings
if($field[4] < 100) {$field[4]=100;}
```

Proposed addition for recording the Lat Long for ePingER

We propose adding the Lat Long to the end of each (more recent) record sperated by a semi-colon (;).

Src_name	Src_addr	Dst_name	Dst_addr	Len	Unix_GMTs	sent	rcv														
min	avg	max	seqs	RTTs				Lat	Long												
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	37.4471	-122.1755											