PlanetLab ruby script

```
script:#\! /usr/local/bin/srinterpreter

Destination = "134.79.18.188"
Probes = ARGV.length>1 ? ARGV[1].to_i : 10
AvgIntervalSec = 10.0/Probes.to_f; # hmm, didn't get this right.
xprobe = Scriptroute::Icmp.new(16)
xprobe.ip_dst = Destination

packets =
Scriptroute::send_train( ( 1..Probes ).map { |rep|
                           probe = Scriptroute::Icmp.new(16)
                           probe.ip_dst = xprobe.ip_dst
                           probe.icmp_type = Scriptroute::Icmp::ICMP_ECHO
                           probe.icmp_code = 0
                           probe.icmp_seq = rep
                           Struct::DelayedPacket.new( (rep>1) ? -Math.log(rand)*AvgIntervalSec : 0,
                                                      probe ) } )

packets.each { \|tuple\|
if( tuple.response ) then
response = tuple.response.packet
rtt = (response) ? ((tuple.response.time - tuple.probe.time) * 1000.0) : '*'
if tuple.response.packet.icmp_type \!= Scriptroute::Icmp::ICMP_ECHOREPLY then
puts "Received: "   tuple.response.packet.to_s
else
puts tuple.response.packet.ip_len.to_s   ' bytes from '
tuple.response.packet.ip_src
': icmp_seq='   tuple.probe.packet.icmp_seq.to_s
' ttl='   tuple.probe.packet.ip_ttl.to_s
' time=%5.3f ms' % rtt
end
else
puts "To #{xprobe.ip_dst} timed out"
end
}
```

credentials:251;v1;kalim@slac.stanford.edu;134.79/16;1256421401;(s
#32FFDF5C5DCB7F8F8149EF12532E244548A3166D0DA478888B05ABE1CA654CE0#)