Carnegie Mellon
Computer Science Department.
15-744 Spring 2008 Tools Problem Set 2

This problem set has 2 questions with many sub-parts. Answer them as clearly and concisely as possible. You may discuss ideas with others in the class, but your code, solutions and writeup must be your own. If you do discuss at length with others, please mention in your solution for the problem who you collaborated with. Do not look at anyone else's solutions or copy them from anywhere.

This assignment is due by **4:30pm, Friday, April 11th** to the course secretary in Wean Hall 8018, or you can slip it under Vijay's door in WeH 8101.

Glossary

ICMP: The Internet Control Message Protocol. The protocol that ping uses.
Ruby: A cool scripting language. You'll be modifying an existing ruby script for this assignment.
RTT: Round-Trip Time.
CDF: Cumulative Distribution Function.

1. Scriptroute is a network measurement tool that makes it easy to write network measurement scripts and to perform distributed measurment by running those scripts on remote servers.

   First, take a peek at the Scriptroute page:

   `http://www.scriptroute.org/`

   To compile from source, grab

   `http://www.scriptroute.org/source/scriptroute-0.4.8.tar.gz`

   configure: (this works on linux.gp cluster machines)

   ```
   ./configure --prefix=/YOUR/HOME/DIR/scriptroute
   make
   make install
   ```

   Then, try running a ping from the www.scriptroute.org node:

   ```
   cd /YOUR/HOME/DIR/scriptroute/bin/
   edit sr-remotely.rb
   and add
     $LOAD_PATH << ''/YOUR/HOME/DIR/scriptroute/lib/ruby/1.8''
   at the top.

   Then, run

   ./sr-remotely.rb www.scriptroute.org sr-ping www.cs.cmu.edu
   ```

   Executing sr-ping will take about 10 seconds, so be patient. At the end, you should see the ping output. (Note that you don't have to do this testing through sr-remotely; if you have your own machine on which you can install the scriptroute program as root, go ahead and do all these things locally.)

(a) Modify the `sr-ping` script so that it takes two new arguments: The second should specify an integer number of packets to send, and the third should specify the amount of time to wait in-between packet transmissions. The script should default to 10 and 1, respectively.

(b) Modify the `sr-ping` script so that it also prints out the IP ID of the ICMP echoreply. For example:

```
> ./sr-remotely.rb www.scriptroute.org sr-ping www.cs.cmu.edu 3
host: www.scriptroute.org
file: sr-ping
args: www.cs.cmu.edu 3

1 128.2.203.164 (128.2.203.164) 0.227 ms [id=16109]
2 128.2.203.164 (128.2.203.164) 0.349 ms [id=16110]
3 128.2.203.164 (128.2.203.164) 0.169 ms [id=16111]
```

(c) Execute `sr-ping 128.30.76.114 3 5`. This will send an ICMP echo request every five seconds. Include the output of this program. How is the IP ID field being incremented? (Hint: This might be easiest if you run this experiment at night or during off-hours.)

Now, repeat the above command but *also* run a normal ping simultaneously to the same machine. Include the output. What's different? Why is the output different?

(d) Modify sr-ping to take a third (and final, we promise) argument, `-tcp`. If this argument is specified, sr-ping should send probes using TCP SYN packets to port 8000 of the specified machine. (hint: see the `sr-tcptraceroute` file...)

(e) Find a server that responds to both ICMP and TCP pings. Record the ping times for 600 pings using both ICMP and TCP. Send pings no more than two per second (delay of 0.5). Note that if you're executing scriptroute remotely, you'll have to use multiple executions of the script—it doesn't want to run for more than 30 seconds. We suggest 10 executions of 60 pings each at 0.5 seconds delay.

Analyze this script to show the min/max/average RTT and the stddev of the RTTs. (If this sounds painful, you might steal Dave's analysis scripts from

`http://www.cs.cmu.edu/~dga/dga-util.tar.gz`

Then plot the CDF of round-trip times for both the TCP ping and the ICMP ping.

What conclusions can you draw from this data? Note that the effects we're looking for may not happen on all paths. If your data is inconclusive, what differencs might you expect to see and why?

(f) Try pinging `www.ebay.com` with normal ping. What happens?

(g) Now try the same experiment, but send 5 pings spaced 1 second apart with your `-tcp` flag in `sr-ping`. What's happening?

(h) Based upon the above, give two reasons why TCP-based probing might be better than ICMP-based probing.

2. Extend your ping program to "round-robin" through a series of comma-separated IP addresses, like

```
sr-ping 3 1 www.cs.cmu.edu www.cmu.edu www.yahoo.com
```

(Note that we've put the server list at the end. In the latest version of sr-remotely, it appears that reverse dns-resolution is performed per-argument, so to handle an arbitrary number of servers, we put the server list at the end. In earlier releases, you could do something like ./sr-remotely remoteServer server1,server2,server3 3 1, but it doesn't properly reverse dns server[x] now.) You will probably have to make small modifications to the sr-ping script to handle this new order.

First, try executing this command:

```
sr-ping 2 5 203.215.155.14 203.215.155.69
```

What appears to be going on with the IP IDs?

Now, try sending TCP probes instead:

```
sr-ping 2 5 -tcp 203.215.155.14 203.215.155.69
```

What's going on with the IP IDs this time? What is a possible explanation for this relationship? Can you think of a useful application for this behavior? (Hint: One of the programs included with scriptroute makes use of this feature.) Why might the router be giving different answers with ICMP and TCP probes? (hint: think fast path vs. slow path processing.)[1]

[1] Interestingly, this behavior appears to be new as of Spring 2007—Dave had never seen it before now. Older routers would respond the same to both ICMP and TCP probes.