EECS 325 P2 Writeup Isaac Ng ikn3

1. **How will you match ICMP responses with the probes you are sending out (list all ways you could think of in your report and use the one that you found to work for you)?**

For me, I checked if the ICMP response matched with the problem by checking that the IP addresses, port number, icmp\_type, and icmp\_code all match up. I got all of these by unpacking from the recv\_packet the various bits that represented these fields and checking them to make sure they equaled the right values/parameters.

Possible Ways to Match ICMP responses:

* + ICMP header’s code and type are both equal to 3
  + IP Header’s protocol field is correct (17 for UDP)
  + Source IP address is the same as the measurement target’s
  + IP Header’s destination address is the same as measurement target’s
  + The destination port number is verified to be the same as the port specified
  + Compare the original payload to the payload sent out to see if it matches
  + Protocol field of ICMP reply’s IP header is the correct IP protocol number for ICMP, aka 1
  + TTL of IP header not larger than 32, our original specified amount

1. **List all possible reasons you can think of for not getting the answer when probing an arbitrary host.**

Possible Reasons:

* Packet gets dropped on route to end destination (high traffic)
* ICMP message gets dropped on way back (high traffic)
* Firewall drops packet (server’s firewall keeps port that we used closed)
* Firewall blocks ICMP packets (therefore, reply gets blocked due to way firewall is configured)
* Receiver drops the UDP segment (Server drops file)
* UDP segment’s RTT is greater than the timeout value we set (will invoke retry)
* The destination port is already in use (port we specified might be in use by another student)
* Too many hops on the path (TTL will decrease to 0 and packet will be dropped)
* No internet connection (LOL, it’s 2017 get some wifi)
* Server is down (somebody crashed the server of google.com or whatever website or even crashes your server)
* Your program isn’t written right (hope you write it right next time and then test it again)

1. **DATA/RESULTS:**

**Sites that worked:**

github.com

hltv.org

teamfortress.tv

google.com

wordpress.com

youtube.com

facebook.com

livejasmin.com

target.com

wiki.com

**Sites that didn't work**:

yahoo.com

amazon.com

twitch.tv

cnn.com

pornhub.com ☹

xvideos.com ☹

stackoverflow.com

bing.com

instagram.com

yelp.com

**Data/Tables/Graphs:**

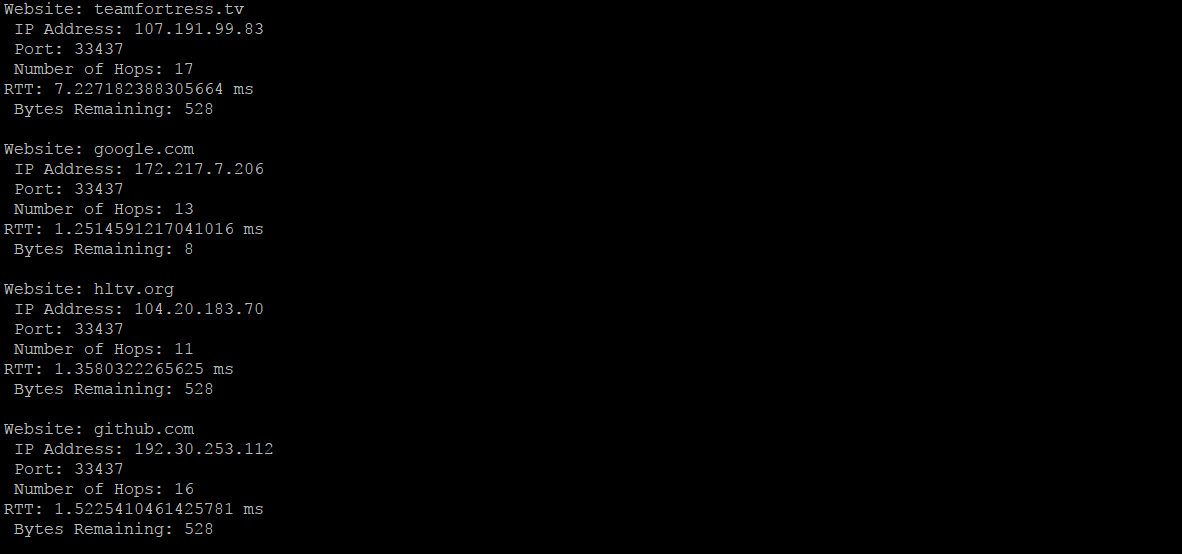
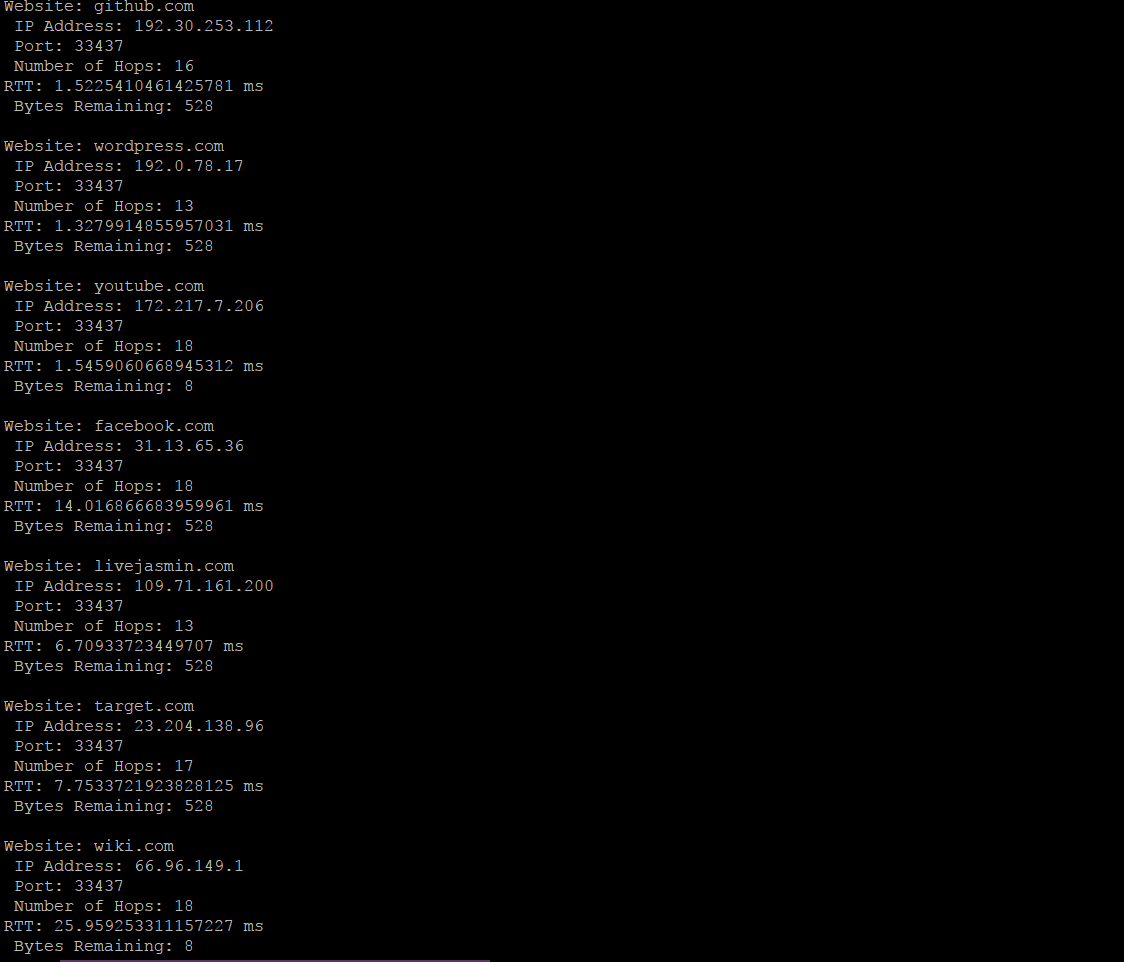


Figure 1: distMeasurement.py data for 10 websites

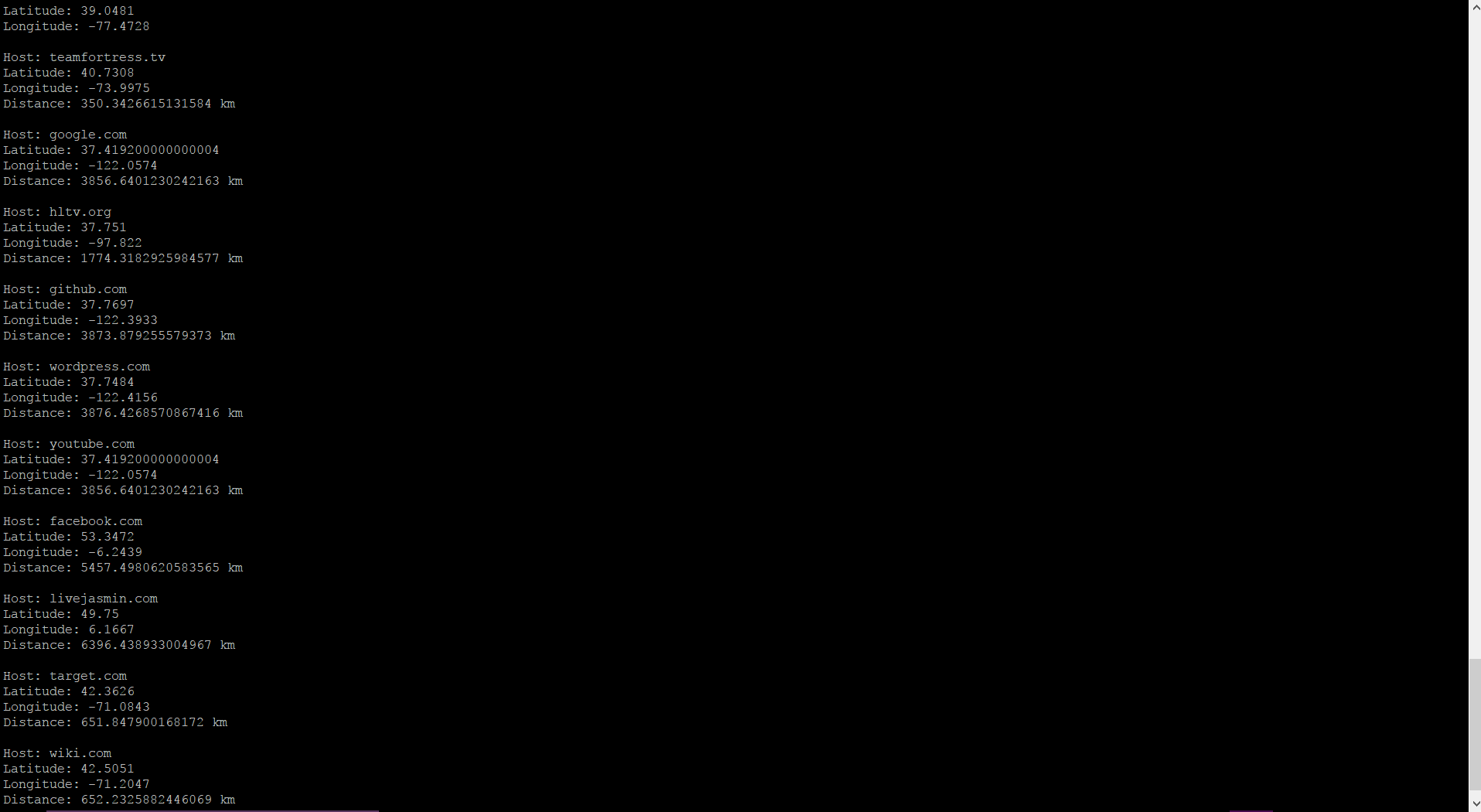


Figure 2: geoDistance.py data for the servers

|  |  |  |  |
| --- | --- | --- | --- |
| **Website** | **RTT (ms)** | **Number of Hops** | **Distance (km)** |
| Wordpress.com | 1.33 | 13 | 3876 |
| Youtube.com | 1.55 | 18 | 3856 |
| Facebook.com | 14.02 | 18 | 5457 |
| Livejasmin.com | 6.71 | 13 | 6396 |
| Target.com | 7.75 | 17 | 652 |
| Wiki.com | 25.96 | 18 | 652 |
| Teamfortress.tv | 7.23 | 17 | 350 |
| Google.com | 1.25 | 13 | 3856 |
| Hltv.org | 1.36 | 11 | 1774 |
| Github.com | 1.52 | 16 | 3874 |

Table 1: Complete Data Table

Graph 1: RTT vs Hops with each Website

\

Graph 2: The Linear Trendline on RTT vs Hops Graph

Graph 3: Number of Hops vs Distance Graph

Graph 4: The Linear Trendline on Number of Hops vs Distance Graph

Graph 5: RTT vs Distance graph

Graph 6: Trendline of RTT vs Distance Graph

I did a graph of the trendline separate from the original graphs because I couldn’t get the linear trendline with its correlation coefficients on the original graphs.

Correlation Coefficients:

RTT and Hops: .30

RTT and Distance: .0814

Hops and Distance: .0564

**Conclusion:**

There is some extremely limited correlation in the first graph with a correlation coefficient of .3, but there is none in the last two graphs. In fact, the linear trendline for the last two graphs show a downward correlation instead of a positive one that it should present. In general, we can expect that RTT does increase with the number of hops and distance. The number of hops should also probably increase with increasing distance. Obviously, there is something going on with the server/other factors that prevent my graphs from showing this correlation. I also noticed my virtual machine is in Ashburn, Virginia (just a cool fact). I think that there is some other factors like congestion that would increase RTT for the same amount of hops and distance. Also, websites that are widely supported like google and youtube will support better link speed and bandwidth compared to wiki.com. Because of this and their infrastructure, they will have lower RTT times than some of the other websites on our list. In addition to this, the various firewalls and proxies could cause the RTT to increase without increasing the distance or hop count. In addition, maybe the server was being contacted at a bad time (network utilization), which could affect the RTT and hop count. In conclusion, we know what to expect for the trends, but due to a number of factors, my graph shows almost no correlation sometimes.