

## Programming Assignment #1 Clarifications

- PartA: Wireshark Lab: ICMP

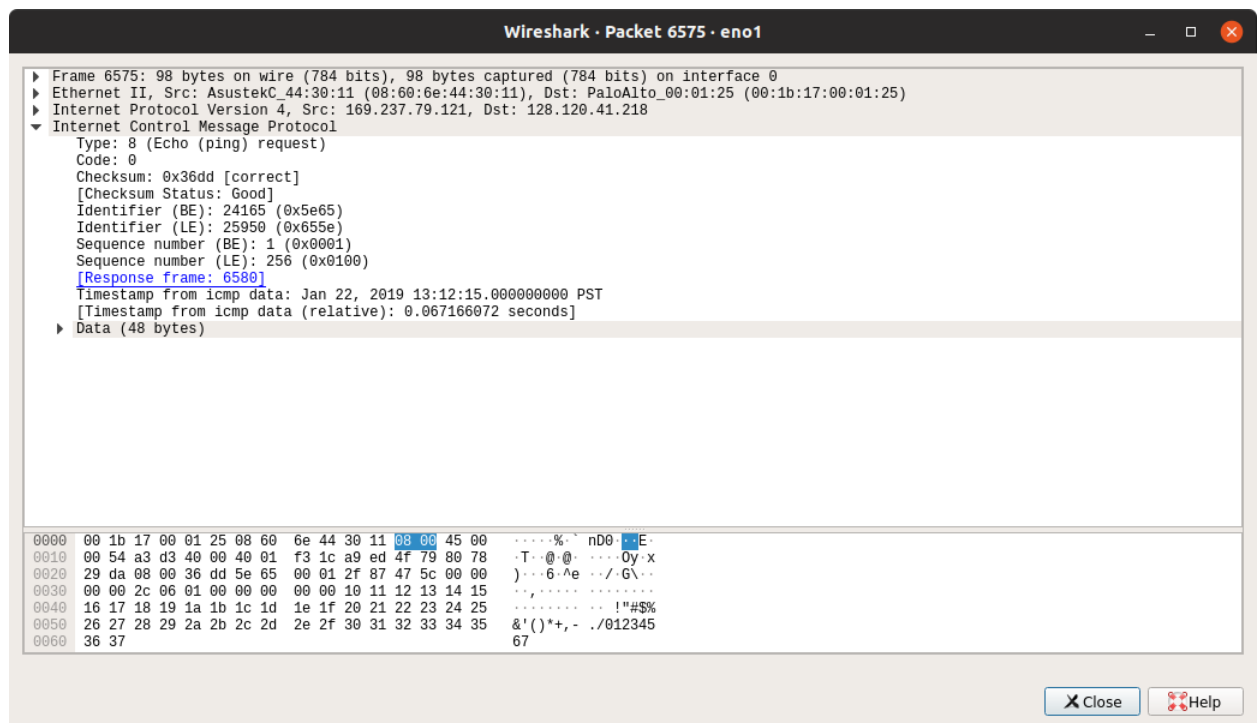
### For PartA.1: ICMP and Ping

1. You may select “ANY” destination you choose, it doesn’t have to be on another continent!!!
2. You may use “ANY” operating systems, windows, Mac, unix/linux, as long as you run the “ping” program on your system and get the result.
3. You need to hand-in a screenshot of the output of your “ping” program, e.g.

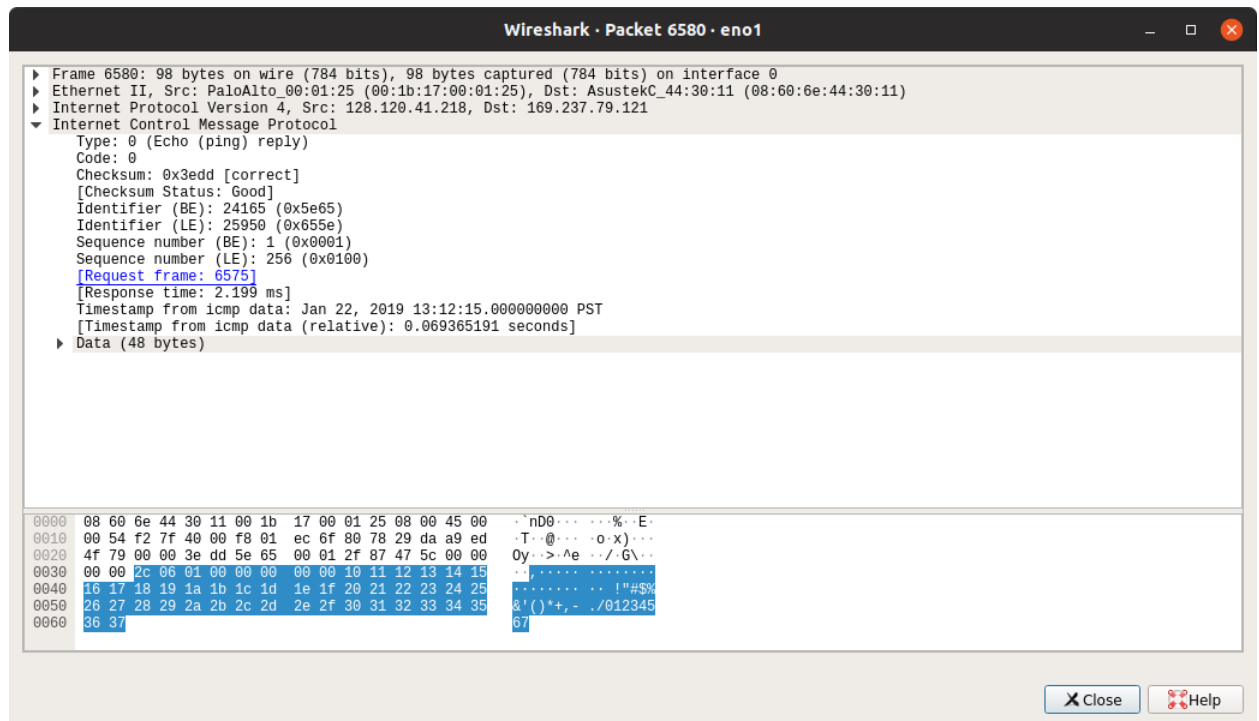
```
chang@chang-server:~$ ping -c 10 my.ucdavis.edu
PING my.ucdavis.edu (128.120.41.218) 56(84) bytes of data.
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=1 ttl=248 time=2.05 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=2 ttl=248 time=2.03 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=3 ttl=248 time=2.05 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=4 ttl=248 time=2.12 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=5 ttl=248 time=2.29 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=6 ttl=248 time=2.09 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=7 ttl=248 time=2.08 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=8 ttl=248 time=2.19 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=9 ttl=248 time=2.15 ms
64 bytes from myservices.ucdavis.edu (128.120.41.218): icmp_seq=10 ttl=248 time=2.12 ms

--- my.ucdavis.edu ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9010ms
rtt min/avg/max/mdev = 2.039/2.122/2.291/0.078 ms
```

4. You need to hand-in a screenshot of an ICMP echo request packet, with ICMP header info explicitly listed, e.g.



- You need to hand-in a screenshot of an ICMP echo reply packet, with ICMP header info explicitly listed, e.g.



- You should answer the questions.

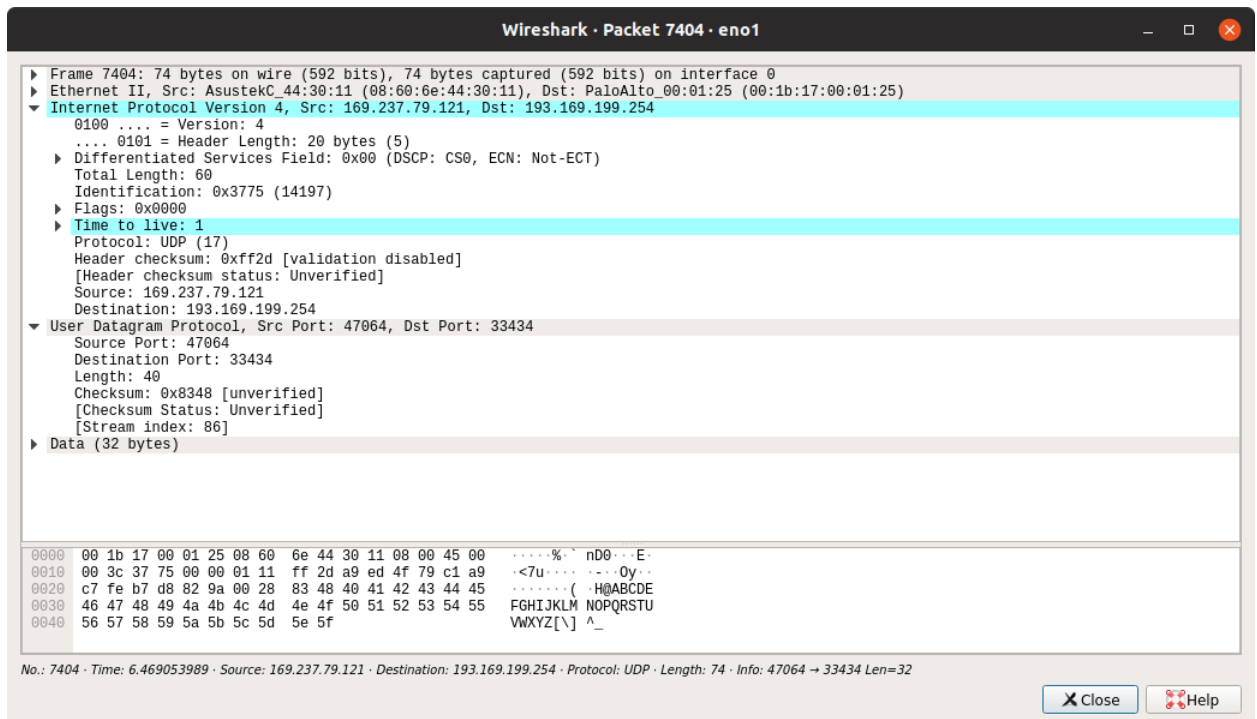
**Overall, you need to hand-in 3 figures and your answers to the questions for PartA.1**

For PartA.2: ICMP and Traceroute

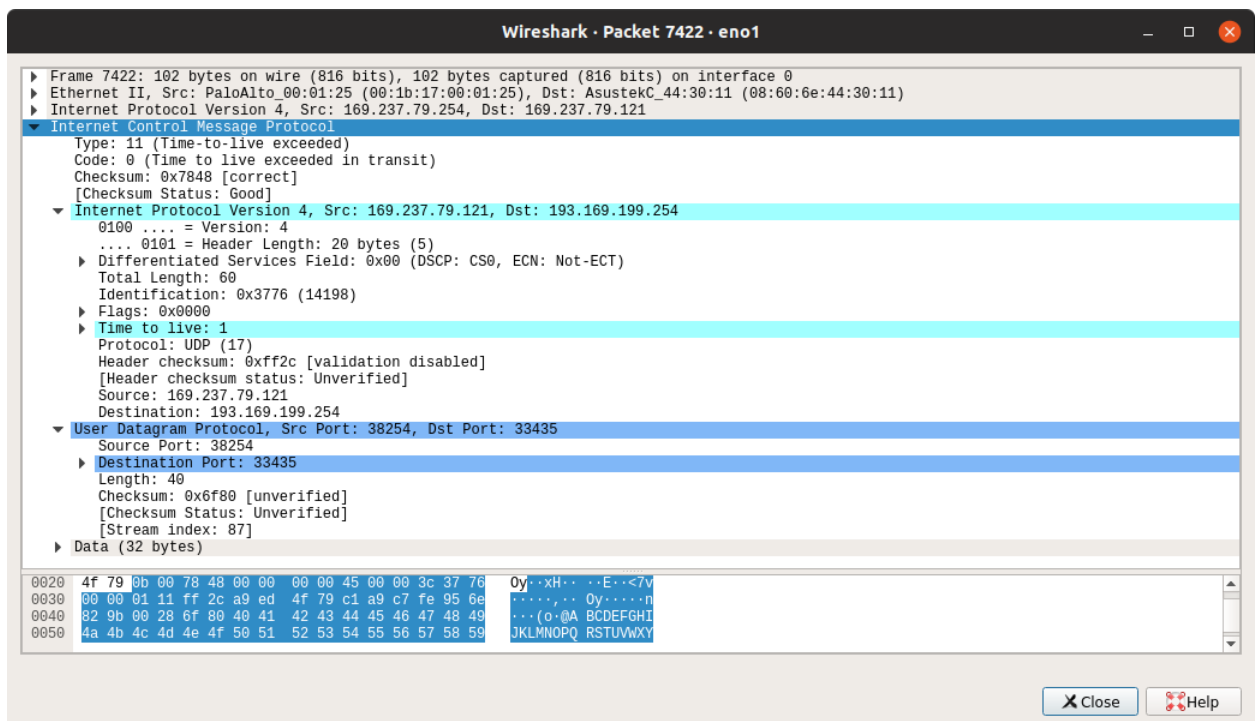
- You may select “ANY” destination that is on another continent (Yes, you will need a host on another continent to see the delay introduced by trans-oceanic link).
- You may use “ANY” operating systems, windows, Mac, unix/linux, as long as you run the “traceroute” program on your system and get the result.
- You need to hand-in a screenshot of the traceroute result, e.g.

```
chang@chang-server:~$ traceroute 193.169.199.254
traceroute to 193.169.199.254 (193.169.199.254), 30 hops max, 60 byte packets
 1 _gateway (169.237.79.254)  1.066 ms  1.022 ms  0.977 ms
 2 169.237.233.17 (169.237.233.17)  1.314 ms  1.149 ms  1.384 ms
 3 bdf2-3--adf2-1.ucdavis.edu (128.120.2.9)  1.117 ms  1.176 ms  1.265 ms
 4 adf2-1--cdf1.ucdavis.edu (128.120.10.133)  1.412 ms  1.173 ms  1.560 ms
 5 cdf1--border-dav.ucdavis.edu (192.82.111.83)  1.838 ms  1.763 ms  1.751 ms
 6 137.164.3.40 (137.164.3.40)  3.648 ms  3.590 ms  3.573 ms
 7 svl-aggr4--oak-aggr4-100ge.cenic.net (137.164.46.40)  4.088 ms  4.183 ms  137.164.46.166 (137.164.46.166)  4.150 ms
 8 10-1-1-91.ear1.SanJose1.Level3.net (4.15.122.45)  4.087 ms  4.334 ms  4.580 ms
 9 ae-1-3.bar1.Sofia1.Level3.net (4.69.143.129)  188.967 ms  188.967 ms  188.942 ms
10 ae-1-3.bar1.Sofia1.Level3.net (4.69.143.129)  194.560 ms  188.900 ms  189.079 ms
11 SPECTRUM-NE.bar1.Sofia1.Level3.net (212.162.46.90)  195.756 ms  189.926 ms  189.747 ms
12 * * *
13 mtel-satcom-inter.mtel.net (213.226.1.1)  182.085 ms  187.092 ms  182.030 ms
14 213.226.7.168 (213.226.7.168)  182.006 ms  190.116 ms  182.063 ms
15 router.bix.bg (193.169.199.254)  189.689 ms  187.012 ms  186.970 ms
```

4. You need to hand-in a screenshot of a probe packet (it maybe be over ICMP/UDP/TCP), explicitly showing the IP header and ICMP/UDP/TCP header if any, highlight the TTL field. e.g.



5. You need to hand-in a screenshot of a ICMP error packet (containing ICMP TTL-exceeded message), explicitly showing the ICMP header fields. E.g.



6. You need to hand-in your answers to the questions.

**Overall, you need to hand-in 3 figures and your answers to the questions for PartA.2**

For PartA.3: Extra Credit

As we haven't cover socket programming, you can ignore the "Extra Credit" section for this assignment.

**IMPORTANT: You need to do your own experiments and get your own results. You won't receive any credits if you reuse any figure in this document!!!**

- Part B: BGP routing prefixes and routing table lookup
  1. For compiled languages such as (C, C++, Java, go), you should hand in your source file and compiled binary. Your compiled binary should be named as "ip2as". For scripting language such as (Python, Pearl), you should hand in your program scripts. And the file should be named as "ip2as" plus language-specific suffix.
  2. For compiled languages, your compiled binary should take exactly two arguments from the command line. TA will run your program as: "*ip2as <DB\_file> <IP\_file>*". For scripting languages, your script should take exactly two arguments from the command line. TA will run your script as: "(e.g. python) *python ip2as.py <DB\_file> <IP\_file>*"
  3. Your program should write results to a file named "output.txt". The output format should look like the following:

```
169.237.0.0/16 1852 169.237.33.90
208.0.0.0/11 1239 208.30.172.70
<ip prefix>/<prefix length> <AS number> <ip from IP_file>
```
- Part C: Using a single computer to probe the internet.

This is an open-ended question. There is no single correct answer. What we are looking for is how you approach the problem and your arguments of why your solution makes sense.