

CSC / ECE 573 Internet Protocols, Fall 2005

Homework #1

[Home](#)[Syllabus](#)[Calendar](#)[Homework](#)[Project](#)[Message
Board](#)[Exams](#)[Links](#)

Due Date

- Tuesday, August 30, at 11:45 PM

Instructions

- Homeworks should be submitted individually. We will use the standard [submit utility](#) for our class to submit all work, which means your work must be prepared electronically.
- Put your name, the assignment number, and date at the top of the first page. Put solutions in order (don't make the TA hunt for your solution).
- Do not plagiarize; that means, do not copy content from any source without permission from the instructor, and if permitted, acknowledge the source.

Problems

1. Compute a checksum as the ones-complement sum of the following 8-bit words #1 through #4, and then ones-complement that sum. Show the 8-bit result.
00100000 + 10001011 + 01001101 + 01000100
Will an error in the rightmost bit of both word #2 and word #3 be detected? How about an error in the leftmost bit of word #2?
2. An IP packet, without options, is to be sent across a link with MTU = 760. Some of the fields of this packet's header, before fragmentation, are shown below. This packet also includes a 8-byte UDP header, immediately after the IP header. Show what these fields are for each fragment, and indicate where and in what fragment(s) the UDP header will appear.
* Header Length = 5, Total Length = 3000, Identification = 25335
* Flags: DF=0, MF=0, Fragment Offset = 0
3. Indicated below is a path through a network from host A to host B. Show what is recorded in the IP Source Route Option header before sending this packet and after receiving this packet if strict source routing is used to follow this path. The path is:
* Host A = 192.168.10.98
* First router incoming interface = 192.168.10.94, outgoing interface = 192.168.22.90
* Second router incoming interface = 192.168.22.86, outgoing interface = 192.168.35.86
* Third router incoming interface = 192.168.35.83, outgoing interface = 192.168.46.80
* Host B = 192.168.46.75
4. Decode the following IP header, shown in hexadecimal, and state clearly (in decimal or in words) what each field value means:
45 00 00 28 c1 e8 00 00 e3 06 49 7b 98 01 02 f4 98 0e 3e 69
5. Layering, like modularity, is a way for people to deal with complexity, and as such seems like a good idea. What is an example of a networking function where you might want or need to violate the layering principle? Justify your answer with a few sentences.
6. Do a little web research, and find what % of IP datagrams are fragmented in practice. Cite your source. [CAIDA](#) tends to be a good first place to look for data about the Internet.
7. Using http://www.ietf.org/iesg/1rfc_index.txt, create a table showing how many RFC's were adopted in the 60's/70's (combine these), the 80's, the 90's, and so far in the 00's. (Don't worry about missing or unused numbers). What do you conclude about the rate of standards development?

8. RFC 1958 ("Architectural Principles of the Internet") is an entertaining and opinionated compilation of IETF "philosophy". One statement from there is "Be strict when sending and tolerant when receiving". What does this mean, and are there any undesirable consequences of adopting this policy in implementing protocols?
9. I ran "netstat -s" on machine remote-linux.eos.ncsu.edu, and got the following partial output:

```
Ip:
  1954212 total packets received
  0 forwarded
  0 incoming packets discarded
  936300 incoming packets delivered
  1402038 requests sent out
  1020093 reassemblies required
  276206 packets reassembled ok
  27989 fragments received ok
  107619 fragments created
Icmp:
  2941 ICMP messages received
  0 input ICMP message failed.
ICMP input histogram:
  destination unreachable: 65
  timeout in transit: 1
  echo requests: 2857
  echo replies: 18
  2857 ICMP messages sent
  0 ICMP messages failed
ICMP output histogram:
  echo replies: 2857
Tcp:
...
Udp:
...
TcpExt:
...
```

From this data, does it appear that fragmentation occurs very frequently?

10. From your computer, use the "tracert" command ("traceroute" on linux) to find the longest path (largest number of hops) you can to any IP address in the world (be creative :-). What is this destination, and how many "hops" did it take to reach this destination from your machine? [Note: geographic distance does not always correlate with number of hops.]
11. Is there anything wrong with multiple IP addresses mapping to the same MAC address on an Ethernet? Why or why not?
12. Is there anything wrong with multiple MAC addresses on an Ethernet mapping to the same IP address? Why or why not?
13. On the Homework website is a [dump file](#) captured by ethereal. Download and install ethereal on your own machine and process this file to answer a few questions. [The Ethereal "Help" function by the way is not very good. I recommend instead looking at the online documentation at <http://www.ethereal.com/docs/>.] [Note 2: If you click "Expression" beside "Filter", you'll get a list of the keywords that can be used in filters.]
- * How many packets are in this dump file? (use "Statistics"->"Summary")
 - * How long a period of time does this dump represent?
 - * How many are IP datagrams?
 - * How many are DNS messages?
 - * How many are ARP packets? Why are there so many ARP Requests in this dump, but so few ARP Replies?
 - * Sort by protocol (just click on the "Protocol" column), and list all the protocols you find in packets sent during this short window of time. (There's a lot going on in networks, isn't there?)
14. Take a look at http://www.rootsecure.net/content/downloads/pdf_downloads/arp_spoofing_intro.pdf. Using ARP, what are two ways a host X attached to an Ethernet switch can make sure it receives all Ethernet frames sent by another host Y attached to the same Ethernet switch?
15. The linux networking source code can be found at <http://lxr.linux.no/source/net/>, which links together source files in a convenient-to-browse way. Files specific to IPv4 are in folder ipv4. The source code is difficult to read line for line unless you really study the entire networking subsystem thoroughly. Nevertheless, I'd like for you to get comfortable looking for things in source code, and at least scanning for answers to specific questions. Attempt to answer at least two of the following questions; ask me, the TA, or another student for help if you need it.
- * If multiple replies to ARP request, which one used? [see `arp.c`]
 - * If option should not be included in successive fragments, what does linux do? [see [ip_options.c](#)]

- * What happens if fragment reassembly times out? [ip_fragment.c]
- * What size are fragments? [ip_output.c]

Created on August 22, 2005

Last Modified September 21, 2005

Maintained by [Douglas S. Reeves](#)