

1. **IP Addressing**

Suppose a router has the following routing table:

64.20.44.0/24	if1
64.20.44.100/30	if2
64.20.32.0/20	if3
64.20.46.0/23	if4
64.0.0.0/10	if5
0.0.0.0/0	if6

- (a) For each entry in the table, list the range of IP addresses that will be matched by that entry (ignoring any overlap between entries) and state the number of addresses in each group
- (b) Which interface will be used for each of the following addresses? Remember that routers use the rule with the longest matching prefix.
- 64.20.44.102
 - 64.20.45.102
 - 64.20.47.102
 - 64.20.48.102

2. **TCP RTT Estimation** One difficulty with the original TCP SRTT estimator is the choice of an initial value. In the absence of any special knowledge of network conditions, the typical approach is to pick an arbitrary value, such as 2 seconds, and hope this will converge quickly to an accurate value. If this estimate is too small, TCP will perform unnecessary retransmissions. If it is too large, TCP will wait a long time before retransmitting if the first segment is lost. Also, the convergence might be slow.

- (a) Choose $\alpha = 0.75$ and $SRTT(0) = 2$ seconds, and assume all measured RTT values = 1 second with no packet loss. What is $SRTT(15)$? Recall,

$$SRTT(k + 1) = \alpha * SRTT(k) + (1 - \alpha) * RTT(k + 1).$$

- (b) Now let $SRTT(0) = 1$ second and assume RTT values = 2 seconds and no packet loss. What is $SRTT(19)$?

3. **IP Fragmentation**

Consider two hosts, A and B, each on a separate shared Ethernet with MTU=1500 bytes. In addition to these LAN's, the route connecting host A to host B through the Internet contains an additional hop over a point-to-point link between a router on A's Ethernet and a second router on B's Ethernet. The point-to-point link has MTU=400 bytes. Recall that MTU is the maximum amount of data that can be sent in a frame at the physical layer and thus includes all TCP and IP headers (each of which occupies 20 bytes). Also recall that IP fragmentation breaks data along 8 byte boundaries.

- (a) For each packet that arrives at host B, show the part of the header that concerns fragmentation, i.e. the ident, more-fragments, and offset fields, as well as the payload length.
- (b) How many bytes were sent by host A? How many bytes were received by host B? Include in your calculations the TCP, IP, and Ethernet header sizes.
- (c) If each packet across the point-to-point link arrive intact (without error) with probability p , calculate the probability that the entire sequence of packets arrives without the need for retransmission.
- (d) Calculate the long-term expected efficiency of the point-to-point link, assuming that the probability of each fragment being lost is $4/5$. Include in your calculation the overhead of TCP and IP headers, as well as any retransmissions.

- (e) Most IP datagram reassembly algorithms have a timer to avoid having a lost fragment tie up reassembly buffers forever. Suppose a datagram is fragmented into four fragments. The first three fragments arrive, but the last one is delayed. Eventually the timer goes off and the three fragments in the receiver's memory are discarded. A little later, the last fragment stumbles in. What happens to this last fragment at the receiver?
4. Networking Utilities: `ifconfig` and `arp` Show the commands that you use to solve the problem and the output you get. No credit if you don't show your work. These utilities are located in `/sbin`, which is likely not in your path, so invoke them as `/sbin/ifconfig` and `/sbin/arp`
- (a) The `ifconfig` utility is used to assign and examine network interface parameters. Read the man page on `ifconfig` to determine the Internet broadcast address for the EWS machines. (Hint: Use `ifconfig` with the `-a` option to list output for all interfaces, or first use the `netstat` utility discussed in problem set 2 to determine the specific interface(s) of the host to the network, so that you can specify them in the `ifconfig` command.)
- (b) The `arp` utility can be used to display the Internet-to-Ethernet address translation table in the host it is run on. Consult the man page for `arp` and use `arp` to determine the Ethernet addresses of 5 machines in the `ews.uiuc.edu` domain. (Hint: Since entries in the address translation table time out, there may not be many entries in it. To fill in a lot of entries, send a single ping to the broadcast address you found in part (a). Your host will look at all the return ping packets to build up its address translation table.)