THE UNIVERSITY OF AUCKLAND

Department of Computer Science
COMPSCI 314 S1 C Assignment 2.
Due Friday 2 April 2004, 4 pm.

- This assignment will contribute $^{5}/_{30}$ of the coursework mark and 5% of the overall course mark.
- The prefix M means Mega or 10^6;
  bps is bits per second
  Bps is byte per second
- "User data rate" means the number of bytes that a user sees sent or received in one second.

Q 1 A user message is to be transmitted over a 100 Mbps Ethernet LAN, using TCP (transport layer), IP (network layer).

- Each user message is a single TCP "segment", with a TCP header of 20 octets.
- Each segment (including its header) is carried as one or more IPv6 packets or "datagrams" (but as few as possible).
- Each IPv6 packet has a 40 octet header and may be as large as the physical medium will carry. (1500 octets is the maximum size of an IP packet *including* any IP headers.)
- Allow for the preamble and all other parts of the Ethernet frame, but ignore the inter-frame gap.

◇ *For now the only thing you need to know about IPv6 is that each packet has a 40 octet overhead Similarly, each TCP segment has a 20 octet overhead, and may be "fragmented" among several IP packets.*

◇ *The TCP, IP and Ethernet layers form a layered protocol stack, as discussed in lectures, with each adding a header (and possibly trailer) as the message passes down from the user, for transmission. While you are welcome to read ahead to really understand what is going on, the description given here is adequate for the present.*

(i) Estimate the user data rate for user packets of 20, 200, 2000 and 10,000 bytes. **[6 marks]**

(ii) If each transmitted TCP segment is itself acknowledged by another TCP segment (header only, with no data), estimate the user data rates for user messages of 500 and 10,000 bytes. Assume that each user message is sent as a sequence of segments and packets, and that the sender then waits for an acknowledgement before starting the next user message.

◇ *Using a LAN means that we can ignore propagation delays.*

[4 marks]

(iii) *Comment only* (no full calculations) on the probable effect on user data rate if the IPv6 traffic must at some stage be "tunnelled" through an IPv4 link (IPv4 header = 20 octets).

[2 marks]

Q 2 It is desirable in data communications to avoid character-by-character processing of messages wherever possible, including message copying and, unfortunately, programmed calculation of checksums. This question tries to show you why!

Assume here that each TCP segment (with its header) is first checksummed and then divided into IPv6 datagrams of an appropriate size (again each with a header). All of the IP datagrams for the segment are sent in sequence and the whole segment is acknowledged by a similar reply segment before the next segment is sent. (The user message must be verified as correct before it is acknowledged; if the received segment is 1160 octets, then the acknowledgement will also use a segment of 1160 octets.)

Each checksum or copying operation must be completed before proceeding to the next stage of processing.

The situation is similar to that of Question 1 above, with a 1400 byte user message which will be transmitted over Ethernet as an IPv6 datagram.

Memory bandwidth overheads.

Some operations must process the entire message, traversing it byte by byte (or possibly word by word). Assume the following times —

- checksums, where used, take 100 ns per 2 bytes (16 bits),
- copying a message memory to memory takes 200 ns for each 4 bytes (32 bits). (Where possible this copying should be replaced by data structures and pointers into the original data area, so that a byte is transmitted from the original user data area.)
- copying from memory when sending the Ethernet frame is performed "invisibly" by hardware and has no extra overhead; the same applies to the calculation of the Ethernet checksum.
- The final Ethernet transmission has zero cost, either for memory transfers or for checksum calculation.

Possible memory intensive operations include —

a. Calculate the TCP checksum – *this must done for every segment*

b. Copy the TCP segment into the IP packet – *this may be done for every segment*

c. Copy the IP packet into the Ethernet frame – *this may be done for every packet*

(i) The results may differ by as much as 10% from the correct values. What simplifications does this allow in the calculations (or what quantities may be omitted)? **[2 marks]**

Apply these simplifications to the remaining answers.

deleted ~~**[2 marks]**~~

(iii) Assuming that there are no memory bandwidth overheads at all, calculate the time to transfer a single user segment, for an Ethernet speed of 100 Mb/s, and hence the "user transfer rate" in this case (the number of bytes transferred per second as seen by the user).

[5 marks]

(iv) Calculate the user data rate with different degrees of copying. Assume that there is no copying for physical transmission. Also assume that there is no overlap between the stages of the processing, so that for example all of the IP datagrams must be constructed before the first one is processed.

Consider two cases —

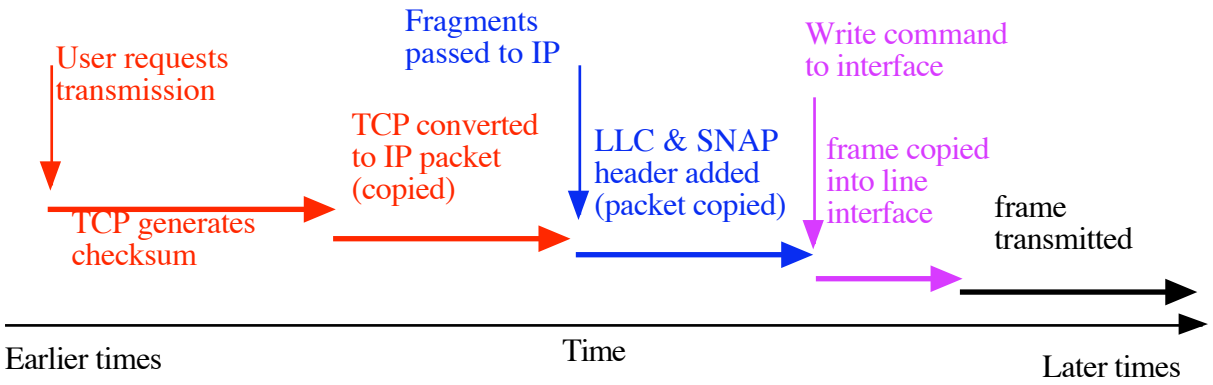
1. The message is checksummed before transmission and after reception, with minimal copying. Transmission may be overlapped with other processing, but a frame cannot start until its IP packet is completely processed.
2. As well as checksumming, the SDUs at each level are copied first by the Transport layer to form its Transport-PDU, and then by the Network Layer to give the Network-PDU. There is similar copying at the receiver to recover the user data. **[6 marks]**

[Assignment total = 25 marks]

Example of time sequencing

This diagram shows the time sequence of operations in the worst case., with the successive operations of calculating the checksum, converting the TCP segment to the IP packets, adding the LLC and SNAP header to the IP packet to get the IEEE 803.2 (perhaps not needed here?) data, possibly copying the frame into the line interface, and finally transmitting the bits across the Ethernet link. (Ignore Ethernet contention.)

Many of the copying steps (*but not the TCP checksum calculation*) can be eliminated with clever data structures that allow portions of user data to be copied from main memory and interleaved with the IP and other headers.



Extreme example of data copying during transmission

Example of message expansion. (Best viewed on the Web PDF file for colour display).

The processing occurs in several steps –

1. The initial user message has a TCP header added to form a TCP segment
2. The TCP segment is split into 3 fragments (one includes the TCP header); each fragment has an IP header added to form an IP packet. (Actual cases may have fewer or more packets in a segment.)
3. Each IP packet is then surrounded by an Ethernet frame (possibly including SNAP header) for physical transmission.

- The diagram is laid out so that a user byte in the user message is (more or less) directly above that same byte in the IP packet and Ethernet frame.
- Corresponding packets and frames are shown in colour.
- The final transmission shows the packets, with headers and trailers, sent in sequence.