ETSF05/ETSF10 Higher layers and QOS tutorial answers

K&R Review questions Chapter 2:

R1
The Web: HTTP; file transfer: FTP; remote login: Telnet; e-mail: SMTP; BitTorrent file sharing: BitTorrent protocol

R2
Network architecture refers to the organization of the communication process into layers (e.g., the five-layer Internet architecture). Application architecture, on the other hand, is designed by an application developer and dictates the broad structure of the application (e.g., client-server or P2P).

R3
The process which initiates the communication is the client; the process that waits to be contacted is the server.

R4
In this case, client and server are roles, depending on the session the peer is involved in. Nothing prevents a peer to be both a client and a server simultaneously, for instance, as it downloads a file while also uploading some of its pieces (as in BitTorrent).

R5
The IP address of the destination host and the port number of the socket in the destination process.

R6
HTTP is an application layer protocol for transferring documents (i.e., web pages) and the objects referenced therein. The HTTP protocol only provides the functions to request and retrieve files. Another necessary component to realize a Web application is a document format that describes a hypertext. This allows a web browser to understand what objects need to be requested after a web page has been downloaded, for instance, images to be displayed next to some text. The standard format for the Web is the Hyper Text Markup Language (HTML).

R9
SSL operates at the application layer. The SSL socket takes unencrypted data from the application layer, encrypts it and then passes it to the TCP socket. If the application developer wants TCP to be enhanced with SSL, she has to include the SSL code in the application.

R10
A protocol uses handshaking if the two communicating entities first exchange control packets before sending data to each other. SMTP uses handshaking at the application layer whereas HTTP does not.

R12
As the HTTP protocol is stateless, the website needs to explicitly do something to identify users. One way is to use cookies as explained in Section 2.2.4. Cookies allow websites to track users across different sessions (i.e., when the browser is closed and re-opened). If it is enough to keep track of a user within a single session, the website can generate a unique ID at the server side and craft webpages so that such an ID is appended to all subsequent requests (for instance, at the end of the URLs).

R13
Web caching can bring the desired content “closer” to the user, possibly to the same LAN to which the user’s host is connected. Web caching can reduce the delay for all objects, even objects that are not cached, since caching reduces the traffic on links.

R19
MX records are needed as they provide a service-specific DNS reply that applies to email addresses only. The same cannot be accomplished with a CNAME record. For instance, assume you own the yourbusiness.com domain and you want all email addresses in the form user@yourbusiness.com to reach mail.yourbusiness.com, while you want all web traffic to yourbusiness.com to reach www.yourbusiness.com. The only option you would have is to set both mail.yourbusiness.com and www.yourbusiness.com as aliases for yourbusiness.com, but this would not work as the former only hosts an email server and not a Web server.

R20
A recursive DNS query allows the server to issue additional queries to other DNS servers before replying to the client. In contrast, an iterative DNS query allows the server to reply with a referral to another DNS server. The client has to explicitly query the DNS server in the referral to proceed with name resolution.

R21
BitTorrent is much faster than the centralized client-server approach when the total upload capacity of the system (i.e., the denominator of the last term in Equation 2.2) is high. This happens when the sum of the upload rates of individual peers is much higher than us.

K&R Chapter 2 Problems:

P7
As no DNS records are cached at any of the servers, all interactions shown in Figures 2.19 and 2.20 are needed for both a. and b.

a. RTT_l+3xRTTr.

b. The same as point a.

c. RTT_l in both cases.

K&R Chapter 9 Review questions:

R5
Clientside buffering can absorb variations in server-to-client delay. If a particular piece of video data is delayed, as long as it arrives before the reserve of received-but- not yet-played video is exhausted, this long delay will not be noticed. Second, if the server-to-client bandwidth briefly drops below the video consumption rate, a user can continue to enjoy continuous playback, again as long as the client application buffer does not become completely drained.

R6
The use of HTTP over TCP allows the video to traverse firewalls and NATs more easily, which are often configured to block most UDP traffic but to allow most HTTP traffic. Streaming over HTTP also obviates the need for a media control server, such as an RTSP server, reducing the cost of a large-scale deployment over the Internet.

R7
No. On the client side, the client application reads bytes from the TCP receive buffer and places the bytes in the client application buffer.

R12
The four main RTP packet header fields are the payload type, sequence number, timestamp, and source identifier fields.

K&R Chapter 9 Problems:

P1
a) Client begins playout as soon as first block arrives at t1 and video blocks are to be played out over the fixed amount of time, d. So it follows that second video block should be arrived before

time t1 + d to be played at right time, third block at
t1 + 2d and so on. We can see from figure that only blocks numbered 1,4,5,6 arrive at receiver before their playout times.

b) Client begins playout at time t1 + d and video blocks are to be played out over the fixed amount of time, d. So it follows that second video block should be arrived before time t1 +2d to be played at right time, third block at t1 + 3d and so on. We can see from figure that video blocks numbered from 1 to 6 except 7 arrive at receiver before their playout times.

c) Maximum two video blocks are ever stored in the client buffer. Video blocks numbered 3 and 4 arrive before t1 + 3d and after t1 + 2d, hence these two blocks are stored in the client buffer. Video block numbered 5 arrives before time t1 + 4d and after t1 + 3d, which is stored in the client buffer along with already stored video block numbered 4.

d) The smallest playout at the client should be t1 + 3d to ensure that every block has arrived in time.

P6

a) 160 + h bytes are sent every 40 msec. Thus the transmission rate is ((160 + h) x 8)/40 Kbps = (32 + 2h) Kbps

b) IP header: 20 bytes UDP header: 8 bytes, RTP header: 12 bytes
h = 40 bytes. Time required to transmit = 40 x 40/(160 + 40) msec = 8 msec.

P10

V oIP can support delays up to about 400 msecs, although a more satisfying conversational experience is achieved with smaller values of q. On the other hand, if q is made much smaller than 400 msecs, then many packets may miss their scheduled playback times due to the network-induced packet jitter. Roughly speaking, if large variations in end-to-end delay are typical, it is preferable to use a large q; on the other hand, if delay is small and variations in delay are also small, it is preferable to use a small q, perhaps less than 150 msecs.

P13

Refer to Figure 9.12. A sends B an INVITE message, which resembles an HTTP request message. This INVITE message is sent over UDP to the well-known port 5060 for SIP. (SIP messages can also be sent over TCP.) The INVITE message includes an identifier for B, an indication of A's current IP address, an indication that A desires to receive audio, which is to be encoded in format AVP 0 and encapsulated in RTP, and an indication that she wants to receive the RTP packets on port 38060. After receiving A's
INVITE message, B sends an SIP response message, which resembles an HTTP response message. This response SIP message is also sent to the SIP port 5060. B's response includes a 200 OK as well as an indication of his IP address, his desired encoding and packetization or reception, and his port number to which the audio packets should be sent. After receiving B's response, A sends B an SIP acknowledgment message.

P14

a) Each of the other N - 1 participants sends a single audio stream of rate r bps to the initiator. The initiator combines this stream with its own outgoing stream to create a stream of rate r. It then sends a copy of the combined stream to each of the N -1 other participants. The call initiator therefore sends at a total rate of (N-1)r bps, and the total rate aggregated over all participants is 2(N-1)r bps.

b) As before, each of the other N - 1 participants sends a single video stream of rate r bps to the initiator. But because the streams are now video, the initiator can no longer combine them into a single stream. The initiator instead must send each stream it receives to N - 2 participants. The call initiator therefore sends at a total rate of (N- 1)*(N-1)r bps, and the total rate aggregated over all participants is (N-1)r + (N-1)*(N- 1)r = N (N-1) r bps.

c) N *(N-1)r bps

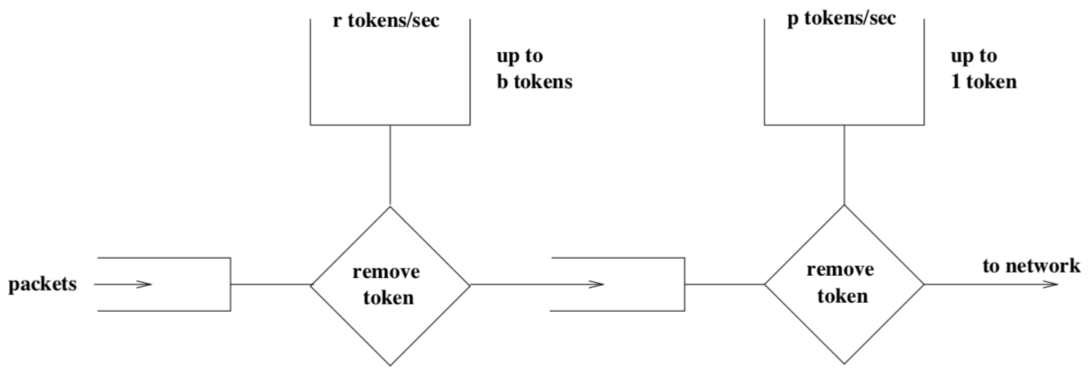
P17 (it should not say leaky bucket but token bucket)

Time Slot	Packets in the queue	Number of tokens in bucket
0	1, 2, 3	2
1	3, 4	1
2	4,5	1
3	5,6	1
4	6	1
5	-	1
6	7, 8	2
7	9, 10	1
8	10	1

Time Slot	Packets in output buffer
0	1, 2
1	3
2	4
3	5
4	6
5	-
6	7, 8
7	9
8	10

P20

See figure below. For the second leaky bucket, $r = p, b = 1$.



Extra questions:

12

Bandwidth, $B = 1\ \text{Tbps} = 10^{12}\ \text{bps}$

Round-trip delay time, $D = \frac{2\times 100\ \text{km}}{200000\ \text{km/s}} = \frac{1}{1000}\ \text{s} = 1\ \text{ms} = 10^{-3}\text{s}$

Bandwidth-delay product, $B \times D = 10^{12}\ \text{bps} \times 10^{-3}\ \text{s} = 10^{9}\ \text{b} = 1\ \text{Gb} = \mathbf{125\ MB}$

13

Since the problem states *"from Europe to US east coast"* and **not** *"from Europe to US east coast **and back**"* we assume that one-way delay time is to be used here instead of the round-trip delay time which is usually used to calculate the bandwidth-delay product.

Bandwidth, $B = 1\ \text{Gbps} = 10^{9}\ \text{bps}$

One-way delay time, $D = 100\ \text{ms} = 10^{-1}\ \text{s}$

Bandwidth-delay product, $B \times D = 10^{9}\ \text{bps} \times 10^{-1}\ \text{s} = 10^{8}\ \text{b} = 100\ \text{Mb} = \mathbf{12.5\ MB}$

If we try to send TCP segments in 1500 byte IP datagrams, the minimum sender window size should be $\frac{12.5\times 10^{6}\ \text{B}}{1500\ \text{B}} = \mathbf{8333\ IP\ datagrams}$ if we want to utilize the link to its fullest.

14

The speed of the airplane and the distance that the airplane has to travel are irrelevant. The bandwidth-delay product is the amount of data "out on the link" from the source to the destination and back. In this case it is:

Bandwidth-delay product, $2 \times \frac{10^{4}\ \text{kg}}{10\ \text{g}} \times 4\ \text{GB} = 2 \times \frac{10^{7}\ \text{g}}{10\ \text{g}} \times 4\ \text{GB} = 8 \times 10^{6}\ \text{GB} = \mathbf{8\ PB}$