2018 Fall Computer Network

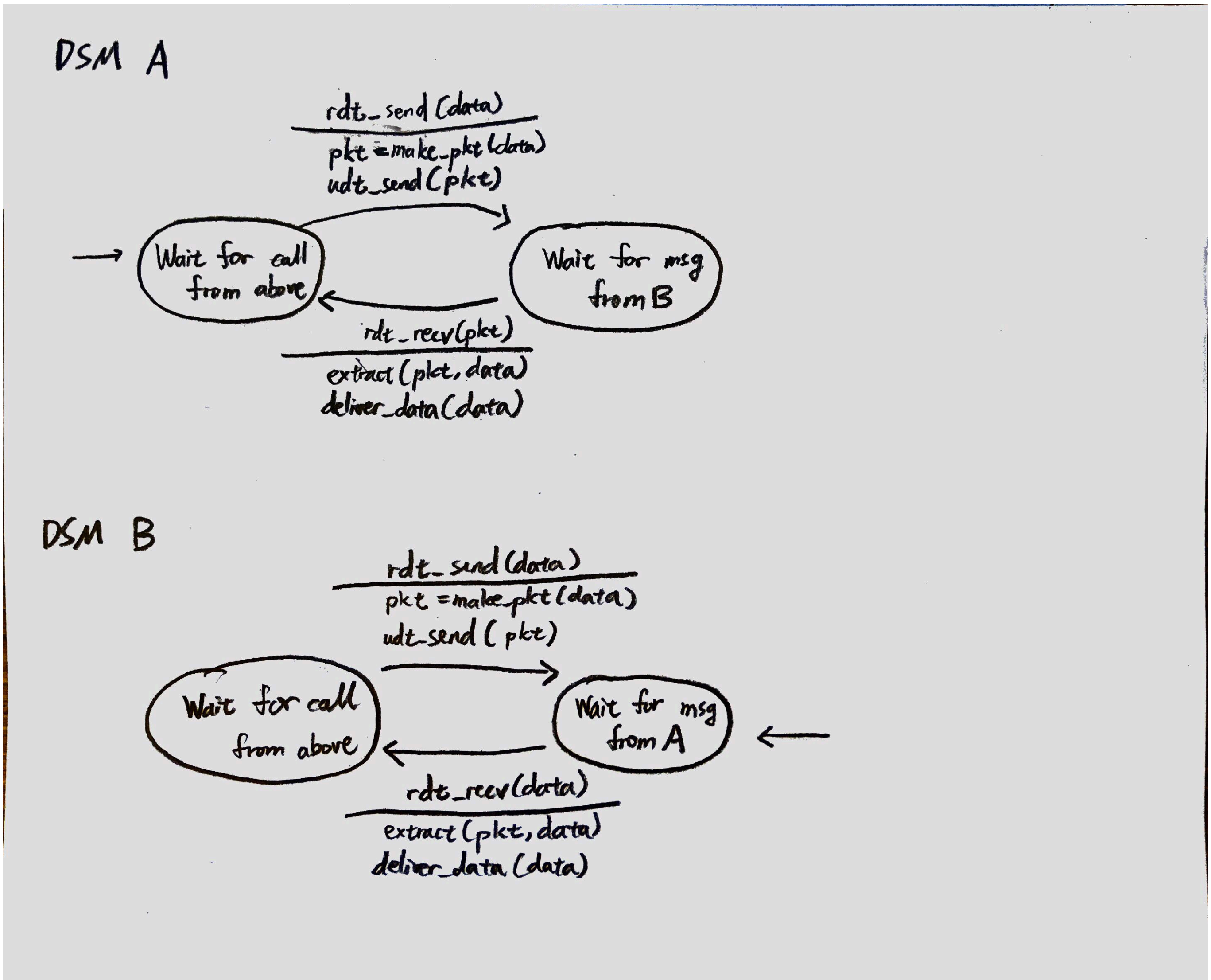
撰寫：資工四 B04705001 陳約廷
日期：2018.12.15
標題：計算機網路 作業四

1. Most firewalls are configured to block UDP traffic, so using TCP can let the traffic through the firewall.
2. Yes, we can design reliable data transfering protocol on application layer to achieve this goal, though this may not be an efficient thing to do because this conflicts with what UDP is originally meant for.
3. If the round trip time is known, the benefit we get is we can know the packet or ACK is lost or not. However in real world situations we would still need a time to estimate these values to detect when packet loss occurs.
4. (a) the first segment contains $110 - 90 = 20$ bytes of data

   (b) TCP acknowledgments are cumulative and hence host B will acknowledge that it has received everything up to and excluding sequence number 90.

5. 2-bit error may go undetected. Suppose we are transfering 2 8-bit bytes. The reciever will check the sum of this two and the 1's complement of the their sum. If the last digit of the 1st word is originally 1 and is flipped to 0. The last digit of the 2nd word is originally 0 and is flipped to 1. In this way the error will go undetected.

6. (a) The reciever expects packet k, so the last 3 ACKs should be $k-3, k-2, k-1$. If none of these ACKs is recieved by the sender, the the window shall be $[k-3, k-2, k-1]$. ACK $k-4$ must be recieved by the sender or else packet $k-1$ will not be sent. If all these ACKs are recieved by the sender, then the window shall go on, and the window shall be $[k, k+1, k+2]$. So all possible windows are: $[k-3, k-2, k-1]$, $[k-2, k-1, k]$, $[k-1, k, k+1]$, $[k, k+1, k+2]$.
   (b) If the receiver is waiting for packet k, that means it has received and ACKed $[k-N, k-1]$ packets. If none of these packets are recieved by the sender, that means ACK messages with value $[k-N, k-1]$ are propagating back. Because the sender has sent packets $[k-N, k-1]$, then it must have received the ACK of packet $k-N-1$, however duplicate ACKs may still be propagating back, so the range of in-flight ACK message values is $[k-N-1, k-1]$.

7. 

8. (a) Slow start is at where window size grows exponentially. So it is interval $[1, 6]$ and $[23, 26]$.
(b) Congestion avoidance is a additive increase on rate. So it is interval $[6, 16]$ and $[17, 22]$.
(c) It is recognized by a triple duplicate ACK, or else the congestion window size would drop to 1.
(d) It is recognized by a timeout, so we can see the congestion window dropped to 1.
(e) The threshold is initially set to 32, because it is when the slow start stops and congestion control begins.
(f) The threshold is set to half the window size when lost is detected, which is $42/2 = 21$.
(g) The threshold is set to half the window size when lost is detected, which is $26/2 = 13$
(h) 1_{st} transmission round: packet 1. 2_{nd} transmission round: packet 2-3. 3_{rd} transmission round: packet 4-7. 4_{th} transmission round: packet 8-15. 5_{th} transmission round: packet 16-31. 6_{th} transmission round: packet 32-63. 7_{th} transmission round: packet 64-96. So the 70_{th} packet is in round 7_{th} .
(i) The congestion value and threshold will both be set half to the current value, so threshold and window will both be $8/2 = 4$.
(j) Threshold set to half of current, which is $42/2 = 21$. The congestion window size is first reset to 1, and then grows to 4
in 19_{th} round.

(k) 17_{th} round: 1 packet. 18_{th} round: 2 packets. 19_{th} round: 4 packets. 20_{th} round: 8 packets. 21_{st} round: 16 packets. 22_{nd} round: 21 packets. So, the total number is 52.