Solution to COMP5311 Assignment 3

QING Pei, 11500811G

November 24, 2011

1 Problem 1

Subnet 1 to 4 will be affected. Subnet 1, 2 and 4 will consider R2 as the next hop for packets to subnet 8. Then the packets will be dropped. Subnet 3 originally uses R3 as the next-hop as R3 reports the distance to subnet 8 is 3, which is less than 4 reported by R1. With the false distance value of R2, R1 will report a smaller distance, 2, to subnet 3, and all the packets from subnet 3 to subnet 8 will be forwarded to R2 and be dropped.

Subnet 5 may be affected, depending on the hosts' decision when R4 and R6 report that they have the same distance to subnet 8. If R4 happen to be chosen as the next hop, for load balancing or other purposes, the packets will be dropped by R2 in the route. If the hosts keep using R6, they won't be affected.

Subnet 6 and 7 remain unaffected since the distance reported by R7, which is 1, is still the smallest, regardless of the false value injected.

2 Problem 2

2.1 case a

After the routing protocol converges, the routing information to destination A is:

At router	Route to A
B	{A,1,B}
C	{A,2,B}
D	{A,2,B}

Immediately after the link A–B breaks, packets with A as the destination

- received by B will be forwarded to C or D
- received by C or D from everywhere but B will be forwarded to B according to the routing table
- received by C or D from B will be dropped as the routing table says the last hop to A is B, which is the same as where the packet is coming from

Therefore the packets will stay alive for at most 3 hops before dropped. No loop is forming.

Later, when B talks with C/D, C/D learns that the route to A with B as last hop is invalid. C and D will stop forwarding those packets to B. They forward to each other and drop the packets as they are from the "last hop in the route".

After that, when C and D talks with each other, they notice that A is actually unreachable and update their routing table accordingly.

From then on, packets with A as the destination received by B, C and D will not be forwarded any further but dropped.

2.2 case b

Immediately after the link A–B breaks, packets with A as the destination

- received by B or E will be forwarded to either C or D
- received by C or D from everywhere but B will be forwarded to B according to the routing table
- received by C or D from B will be dropped

The differences are

- the longest lifetime of packets becomes 4 hops rather than 3 in case a
- C and D does not turn to each other but E after the distance B to A becomes 16
- E will be the last to know that A is unreachable

3 Problem 3

a The destination MAC address is R's MAC address Tp-LinkT_c9:41:05 (00:21:27:C9:41:05). This is acquired from packet 2, telling H1 that the gateway 123.123.2.10 is at that address.

b 123.123.1.10 The router sends this message to tell H1 that subnet 1 should be accessed via that interface (eth1.0) which is also directly connected to H1.

c 2 ICMP headers are included in the IP packet that carries the ICMP Redirect message. The outer header is type 5 (redirect) while the inner one is type 8 or 0 (echo request or echo reply).

d The ARP request for H2's MAC is broadcasted, so H1 will process that packet. The ARP reply is a unicast to H2, H1's NIC will simply drop that packet, and Wireshark won't capture it with the promiscuous mode off.

e The following entry in to H1's routing table at last.

Destination	Gateway	Net mask	Interface
123.123.1.1	123.123.1.1	255.255.255.0	eth0

The MAC address for 123.123.1.1 is updated to be the actual MAC address of eth0 of H2 in the ARP cache at H1. Ping requests to H2 matches the new entry in the routing table and will be sent to Ibm_16:8b:06 (00:11:25:16:8B:06). This does not involve the router.

The echo reply from H2 follow the same rule. Gateway is H2 itself and the MAC address is the real MAC of H1's eth0.

f Packet 8, 12∼15 will not show up in the trace captured at R with the promiscuous mode off. Those packets are not from R. They are neither broadcasts nor unicasts to R's MAC address.

g Packet 1, 2, 3, 4, 9, 10 will not show up in the trace captured at H2 with promiscuous mode off. The src/dest MAC addresses of those packets are either R's or H1's.

h The ICMP headers in packets 3 and 12 in Figure 4 are not identical. The first one tells H1 to use the IP of R's eth1.0 as the gateway. The second one tells H1 to use IP of H2's eth0 as the gateway.

i R will not send ICMP Redirect messages if the hosts are connected to two different physical interfaces. R, as before, forwards the ICMP request from eth1 to some other physical interface to which H2 is connected. Redirect messages are only sent if that interface and H1 are on the same Ethernet segment (connected to the same interface on R). In this case, they are not.
The same to echo reply messages from H2.

4 Problem 4

a $cwnd = 8$ By the time of transmitting data segment 14, there are 8 segments sent unacknowledged. That is equivalent to the window size.

b Window is full and no ACK is received. During the idle period between sending segment 14 and 15, the sender's window is full. Before sending segment 15, sender receives new ACK, so the window shifts by 1 and the window size increases by 1. 2 more segment are now inside the window. Therefore segment 15 and 16 are sent at that moment.

c $cwnd = 15$ At that moment, the total number of data segments sent unacknowledged is 1+14=15 (segment 14, 15∼28).

d During the idle period between sending segment 28 and receiving the 3rd duplicate ACK for segment 13, the window is full. Then the window size changed to $cwnd = 15/2 + 3 = 10$. The window is still full. On receiving each following duplicate ACK for segment 13, $cwnd$ increases by 1. At the moment the 10th ACK for segment 13 is received, the window size is increased to $cwnd = 16$, which is larger than the window size when sending segment 28. The window covers new data and the sender sends that as segment 29.

e $cwnd = 16$ The total number of data segments sent unacknowledged is 1+14+1=16 (segment 14∼29).

f During the idle period between sending segment 34 and 35, the sender's window is full. Before sending segment 35, sender receives new ACK, the fast recovery ends. The window size changes to $cwnd = 7$. Only 6 segments are sent but unacknowledged, so there is one more to send. That is segment 35.

g $cwnd = 7$ The total number of data segments sent unacknowledged is 6+1=7 (segment 29∼35).

h $snd\_max = snd\_una + cwnd * MSS = n + 15m$. $snd\_nxt = snd\_max = n + 15m$

i $snd\_max = n + 15m$ as before. $snd\_nxt = snd\_una = n$ for retransmission.[1]

j Segment 29∼34 would not have been sent before the ACK for segment 28 is received. Those segments were sent because *cwnd* grows larger than 15. If $rwnd = 15$, the window size *cwnd* won't grow over 15. The idle period will continue even if duplicate ACKs for segment 13 are received.
At the moment the ACK for segment 28 is received, $cwnd = 7$, 7 segments (29∼35) will be sent together. Then the sender will become idle, waiting for ACK (new or duplicate).
If the ACK for segment 29 is received, the window shifts by 1 and increases by 1, and 2 new segments are sent. If another new ACK comes, 2 more segments are sent, which is similar to the situation of sending segment 7∼12. This goes on until $cwnd = 15$. Then only 1 segment is sent on arrival of a new ACK.
If duplicate ACK is received, the window size will shrink again. No packets will be sent. Another fast recovery process starts. One segment per duplicate ACK is sent until $cwnd = 15$. Or, a bunch of segments are sent on new ACK.

[1]http://www.ssfnet.org/Exchange/tcp/#c4