Homework 6

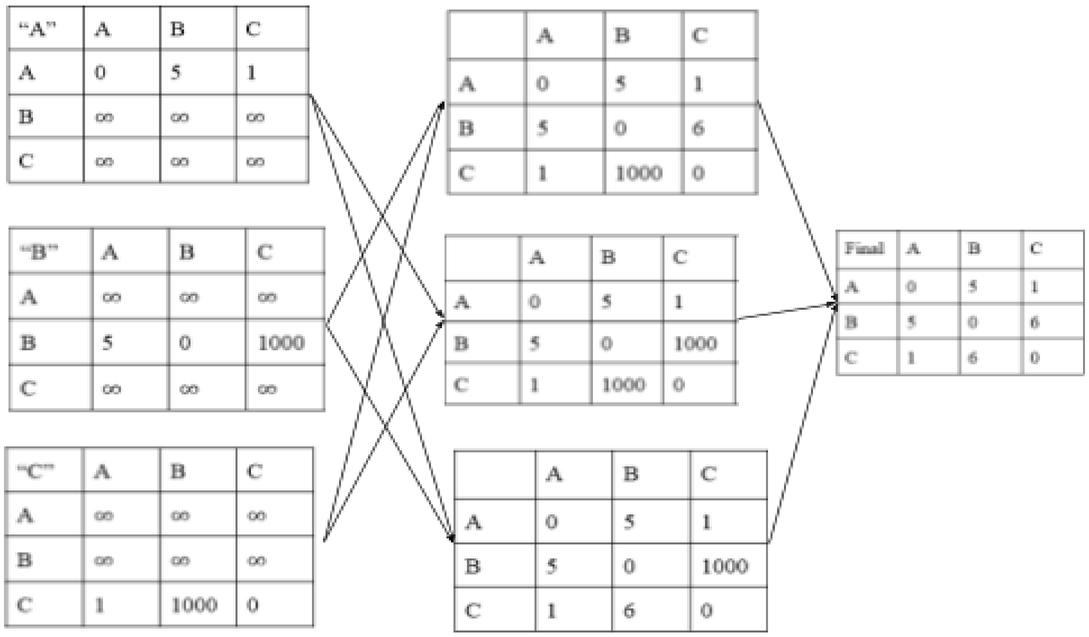
Alex Manasoiu

1. There are 3 subnets with 137.99.17/24 as the prefix, and subnet 1 = 80 interfaces, subnet 2 = 40, and subnet 3 = 40. Since, /24 is the subnet part, it is 32 - 24 = 8 bits for the host. In order to represent this, subnet 1 needs 2^7= 128 addresses since in order to 2^6 = 64 won't be able to support, which ends up translating to 0 - 127 as the host addresses, ending up with 137.99.17.0/25 - 137.99.17.127/25, and the with /25 a the end due to the fact that the prefix range is now 32 -7 bits = 25. For the rest of the subnets we would just need 2^6 = 64 addresses per subnet and the /x will be /26 due to 32 - 6 =26. Due to this the ip address ranges will be 137.99.17.128/26 - 137.99.17.191/26 for subnet 2 and 137.99.17.192/26 - 137.99.17.255/26.

2.

Step	N'	D(s),P(s)	D(t),P(t)	D(u),P(u)	D(z),P(z)	D(w),P(w)	D(v),P(v)	D(y,)P(y)
0	x	∞	∞	∞	∞	1,x	3,x	6,x
1	xw	∞	∞	4,w	∞		2,w	6,x
2	xwv	∞	11,v	3,v	∞			3,v
3	xwvy	∞	7,y	3,v	17,y			
4	xwvyu	8,u	5,u		17,y			
5	xwvyut	6,t			7,t			
6	xwvyts							
7	xwvytsz							

3.
a.



"A"	A	B	C
A	0	5	1
B	∞	∞	∞
C	∞	∞	∞

"B"	A	B	C
A	∞	∞	∞
B	5	0	1000
C	∞	∞	∞

"C"	A	B	C
A	∞	∞	∞
B	∞	∞	∞
C	1	1000	0

	A	B	C
A	0	5	1
B	5	0	6
C	1	1000	0

	A	B	C
A	0	5	1
B	5	0	1000
C	1	1000	0

	A	B	C
A	0	5	1
B	5	0	1000
C	1	6	0

Final	A	B	C
A	0	5	1
B	5	0	6
C	1	6	0

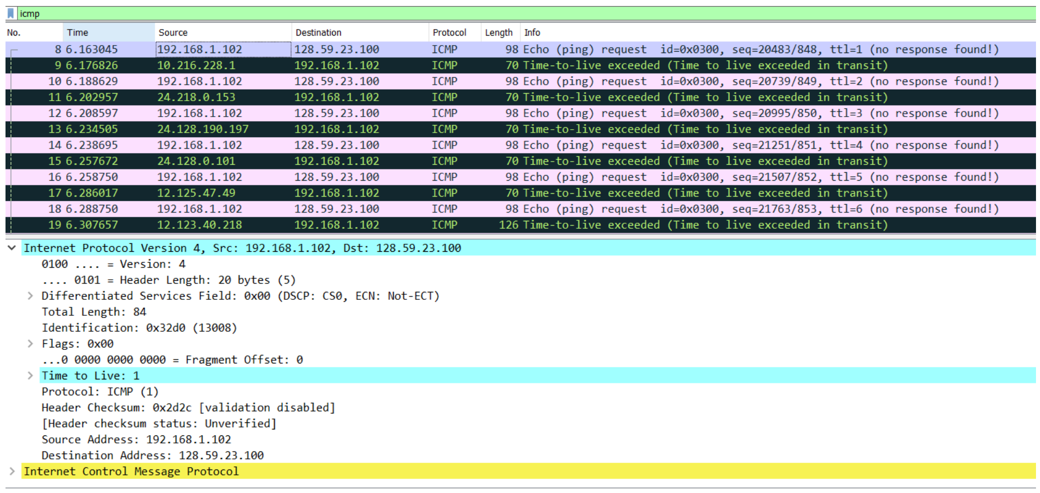
b.
Using the slides, we know that good news travels fast while bad news travels slow, which means that when the link is changed from 5 to 5000, it has to go through a ton of iterations to change it, leading to count to infinity. Originally, we have link $D_C(B) = min(1000 + 0, 1 + 5) = 6$ with the problem of node C thinking A gets to node B with a cost of 5. This means that $D_A(B) = min(5000 + 0, 1 + 6) = 7$, meaning that 1000 - 6 = 994 iteration before stabilization.
c.
Poison reverse is basically the current node telling the next node in line that the current node isn't not connected anymore, meaning that the technique effectively prevents any more packets from being sent through a particular route. This is due to the fact that when the node is "poisoned" the route has become invalid in the network. In our case, if B wants to get to A via C, B will have to say to node A that the B to C route is infinite, effectively blocking A from going through B to get to C. Eventually, once 2 iterations pass, we don't need to do this anymore since the path converges.

4.

	Initial Dist. Vector Table for Node D	
Destination	Cost	Next-Hop
B	12	B
C	1	C
E	2	E

5.

1. Since I was unable to install Traceroute, I used the gaia.umass.edu ip-ethereal-trace1. As a result, it can be seen that the ICMP request was from the computer (source) IP of 192.168.1.102.
2. It seems that the upper layer protocol was ICMP (1)



```
icmp
No.    Time      Source          Destination     Protocol  Length  Info
     8 6.163045  192.168.1.102   128.59.23.100   ICMP        98 Echo (ping) request  id=0x0300, seq=20483/848, ttl=1 (no response found!)
     9 6.176826  10.216.228.1    192.168.1.102   ICMP        70 Time-to-live exceeded (Time to live exceeded in transit)
    10 6.188629  192.168.1.102   128.59.23.100   ICMP        98 Echo (ping) request  id=0x0300, seq=20739/849, ttl=2 (no response found!)
    11 6.202957  24.218.0.153    192.168.1.102   ICMP        70 Time-to-live exceeded (Time to live exceeded in transit)
    12 6.208597  192.168.1.102   128.59.23.100   ICMP        98 Echo (ping) request  id=0x0300, seq=20995/850, ttl=3 (no response found!)
    13 6.234505  24.128.190.197  192.168.1.102   ICMP        70 Time-to-live exceeded (Time to live exceeded in transit)
    14 6.238695  192.168.1.102   128.59.23.100   ICMP        98 Echo (ping) request  id=0x0300, seq=21251/851, ttl=4 (no response found!)
    15 6.257672  24.128.0.101    192.168.1.102   ICMP        70 Time-to-live exceeded (Time to live exceeded in transit)
    16 6.258750  192.168.1.102   128.59.23.100   ICMP        98 Echo (ping) request  id=0x0300, seq=21507/852, ttl=5 (no response found!)
    17 6.286017  12.125.47.49    192.168.1.102   ICMP        70 Time-to-live exceeded (Time to live exceeded in transit)
    18 6.288750  192.168.1.102   128.59.23.100   ICMP        98 Echo (ping) request  id=0x0300, seq=21763/853, ttl=6 (no response found!)
    19 6.307657  12.123.40.218   192.168.1.102   ICMP       126 Time-to-live exceeded (Time to live exceeded in transit)

v Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.59.23.100
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 84
    Identification: 0x32d0 (13008)
  > Flags: 0x00
    ...0 0000 0000 0000 = Fragment Offset: 0
  > Time to Live: 1
    Protocol: ICMP (1)
    Header Checksum: 0x2d2c [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.1.102
    Destination Address: 128.59.23.100
> Internet Control Message Protocol
```

3. There are 20 bytes in the IP header, and 84 bytes total length, this gives 64 bytes in the payload of the IP datagram
4. It does not seem to be fragmented as the flag for fragmenting is 0x00
5. Through all of the ICMP echoes, the identification, time to live and header checksum are always changing.
6. The fields that are staying constant in each ping are version, length, source and destination IP, the ICMP protocol, and the differentiated services field, which means that logically they also are required to stay constant. This is unlike identification, time to live and header checksum, where they are always changing, since they need to change up some things in the ping every time it sends one.
7. The IP header identification keeps incrementing by 1 every ping.