Homework 2

Due: 11 March 2011, 4pm

Note: no late days discounted until **March 14th, 2011 at 4pm**. *However, no points after that, as we will release the solutions by then. If you hand in by the due date, March 11th, 2011 at 4pm, we will grade your homework before the midterm.*

Problem 1 - Addressing, Aggregation, Forwarding [8 pts]

1. You are the CTO of a growing startup and have to get IP addresses to connect 560 computers to the Internet. You can get IP addresses from two providers, IPMart and EastSideIP. IPMart sells classA, class B and class C blocks, while EastSideIP sells CIDR blocks. As the IPv4 address space is scarce, you want to save money and get the smallest number of addresses possible.

a. If you get one block from IPMart, which class do you have to get? What is the problem with that? [1 pt]

> You have to get a class B block from IPMart, because a class C only has addresses for 254 computers. (The first address is the network number, and the last one is the broadcast address.) The problem with getting a class B block is that it allocates a space of 2^{16} addresses, and you only need 560.

b. If you get one block from EastSideIP, how many bits are there in the mask (e.g., is it a /8, /22)? How many addresses are wasted? [1 pt]

> A CIDR block has to have an integral power of two addresses. The smallest power of two larger than 560 is 2^{10}, which means that the network mask has $32 - 10 = 22$ bits. The number of wasted addresses is $1024 - 2 - 560 = 462$. (We also accepted $1024 - 560 = 464$, if you didn't take into account the first and the last addresses of the block.)

c. Suppose you can get two blocks from EastSideIP, and they can be of different sizes. How many bits are there in the masks for each of the blocks? How many addresses are wasted now? [1 pt]

> If we can get two blocks, we should get a /23 block, which will be good for 512-2 = 510 computers. For the remaining 50 computers we need to get a /26 block, good for $2^6 - 2 = 62$ addresses. The number of wasted addresses is 62 - 50 = 12. (We also accepted 16 as the number of wasted addresses, if you considered 512 addresses in the first block and 64 - 48 = 16).

2. Another customer of EastSideIP needs to get 8000 IP addresses. You work for EastSide, and see that you have the address range from 128.140.80.0 to 128.140.112.255 available.

As corrected in the newsgroup, the range is 128.140.80.0 to 128.140.127.255.

a. What is the best CIDR block from this range you can allocate the customer? [1 pt]

128.140.96.0/19 or 10000000.10001100.011/19 , which would be enough for $2^{13} - 2 = 8190$ addresses.

b. Why is it best to minimize the number of CIDR blocks you allocate? [1 pt]

Because the larger the blocks allocated, the smaller the routing tables in upstream routers become.

c. Why is it best to also minimize the size of the address blocks? [1 pt]

Because we improve the utilization of the address space, by reducing wasted allocations.

3. Suppose you have the following routing table in your router

Destination	Netmask	NextHop	Interface
0.0.0.0	0.0.0.0	100.10.1.1	eth0
128.148.0.0	255.255.0.0	128.148.0.1	eth1
128.148.32.0	255.255.240.0	128.148.32.1	eth2
128.148.34.128	255.255.255.128	128.148.34.129	eth3

What is the next hop for each of these addresses, given that you use longest-prefix matching? [0.33 pts each]

a. 128.148.34.143

128.148.34.129

b. 128.148.34.12

128.148.32.1

c. 128.148.38.1

128.148.32.1

d. 200.192.120.12

100.10.1.1

e. 128.148.12.2

128.148.0.1

f. 128.140.0.1

100.10.1.1

Problem 2 - IP Fragmentation [5 pts]

1. You need to send a packet of 4800 bytes over a network path with 3 links A, B, and C, with MTUs of 1500, 576, and 1500 bytes.

a. If the origin doesn't know anything about the path other than the MTU of the first link, how many packets (and their sizes) flow through each link? (Ignore the size of the link layer headers) [1.5 pts]

The original packet is 4800 bytes, which includes an IP header of 20 bytes and a payload of 4780 bytes. (If you assumed that the 4800 bytes did not include the header in the beginning, then your last packet below has an extra 20 bytes. If you were coherent, we accepted this too.)

In the first link, we have to split the payload into k packets that carry at most 1480 bytes (1500 MTU - 20 bytes of header): $4780 = 3 \times 1480 + 340$. So we have 4 packets of 1500, 1500, 1500, and 360 bytes.

The second link needs to split the three 1500-byte packets, which have a payload of 1480 bytes each. The link has an MTU of 576 bytes, which would accommodate a payload of 556 bytes. However, because the fragment offset counts the number of 8-byte groups, the payload must be an integral multiple of 8. $552 = 8 \times 69$ is the largest such multiple smaller than 556. So we will have each 1500-byte packet split into three packets of 572, 572, and 396 bytes, plus the last packet of 360 bytes which won't be re-fragmented, for a total of 10 fragments.

The third link has an MTU larger than the second link, and thus the same 10 packets from link B will go through link C.

b. What fields are different in the IP header for the first fragment that goes on link A and on link B? [1.5 pts]

The total length, TTL, and checksum field will be different. All others will be the same, including the More Fragments flag, the fragment offset of 0, and the IP identification field.

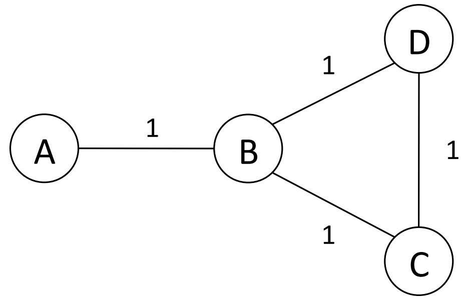
c. If link B has 0.1% chance of dropping a packet (assume the other links don't drop packets), what is the probability that the original packet will be correctly reconstructed in the destination? *(Hint: it might be easier to think that each fragment has a 99.9% chance of making it through B.)* [1 pt]

The probability of the packet being correctly reconstructed is the probability of all 10 fragments making it to the destination, which is $0.999^{10} = 0.990045$.

d. What does this probability change to if the MTU in link B is increased to 1500 bytes? [1 pt]

If B's MTU is increased to 1500 bytes, then we will have 4 fragments, and the probability of success increases to $0.999^4 = 0.996$.

Problem 3 - Distance Vector Routing [7 pts]



Consider network in the graph above, and only consider the routes with destination A. The network is running a distance vector protocol, and considers infinity to be 16. Use the following notation for a routing table entry: B(A,C,1) means that node B has an entry saying it can reach node A through next hop C, with cost 1. Likewise, a route update from B will say B(A,1). Routing table entries time out and are removed if a node does not get an update from the parent after 5 minutes.

a. If link A-B fails, B immediately sets its routing table entry for A to B(A,-,∞). Give a sequence of events in which a loop and count-to-infinity will occur. Suppose nodes are not using any loop prevention technique. [2 pts]

   This is just one example, there are other sequences as well, involving both 2 and 3 nodes:
   C advertises C(A,2) to B, D
   B updates its routing table to B(A,C,3)
   B advertises B(A,3) to C,D
   D updates its routing table to D(A,B,4)
   C updates its routing table to C(A,B,4)
   C advertises C(A,4) to B, D
   B updates its routing table to B(A,C,5)
   In this example a loop forms between C and B, which count to infinity, while D just follows B's count. ...

b. The designers of the protocol decide to add split-horizon to prevent loops. Spoil their party and show a sequence of events that can still cause a loop to form after link A-B goes down, even with split horizon. [2 pts]

   With split horizon, a node never advertises a route to its parent for the route, but it doesn't prevent loops with more than two nodes.
   B advertises B(A,∞) to C,D
   C updates its routing table to C(A,∞)
   D advertises D(A,2) to C (and not to B)
   D updates its routing table to D(A,B, ∞)
   C updates its routing table to C(A,D,3)

C advertises C(A,3) to B (and not to D)
B updates its routing table to B(A,C,4)
B advertises B(A,4) to D (and not to C)
D updates its routing table to D(A,B,5)
D advertises D(A,5) to C
C updates its routing table to C(A,D,6), and so on...

c. In a lot of situations the addition of poisoned reverse to split-horizon does not make any difference, other than increasing the size of the routing announcements. Let's look at a scenario in which there *is* a difference. Suppose the following happens after link A-B fails: 1. B sends a route update B:(A,∞), and at the same time C and D send route updates C:(A,2), D:(A,2). 2. C receives the updates from B and D almost simultaneously, and installs C:(A:D,3) in its routing table. Likewise, D installs D:(A,C,3) in its routing table. Explain how this will evolve if the network is using split horizon, and split horizon with poisoned reverse. [2 pts]

If the network is using split horizon, D will not advertise to C and C will not advertise to D. B will choose one of them as a parent, and advertise a route of distance 3 to the other, which is worse. So C and D will stick to these routes. Since they don't advertise to each other, the routes will eventually time out after 5 minutes with no updates. A large number of packets may loop through the nodes in the meantime.

With the addition of poison reverse, the loop will be broken immediately after the first update from C to D or vice versa, as these will contain a distance of ∞.

d. Explain how a path vector protocol would prevent a loop from forming in this network. [1 pt]

A permanent loop would not form. In a path vector protocol the entire path to the destination is kept and advertised in a route. If a router sees its address as part of a path that it receives from another one, it knows that choosing that route would create a loop.