Ethan Wang
CS 118

CS 118 HW 3

1\. A) If Nethernet requires no minimum padding size, then padding can be removed. Padding is usually needed to reach the minimum packet size for normal Ethernet. Nethernet has no minimum size so we can skip this step now.

B) No, this rule is not valid for Nethernet. Packets can be less than 64 bytes since there is no minimum packet size. Thus, we don't want to just discard small packets since they can hold meaningful information.

C) Nethernet still needs this normal means of detecting collisions because it only focuses on small packet sizes. Collisions with large packets can still occur.
For example, say we have an additional station X in Figure 1. X is located right where the lines intersect for the first time. Both A and B will detect a collision, but X will not. X will only see the collision if it senses the increased voltage. This method of detecting collisions by voltage works well for stations between A and B.

D) It is possible for a station to not detect transmissions if there are no overlapping packets at that station and if it does not initiate a packet transmission. For example, say that we have a station Y that is located directly after station A in Figure 1. There would be no overlapping packets at this point. Also, Y does not initiate a packet transmission, so it will not wait 51.2 usec - the Nethernet method for collision detection fails.

E) Say that station A sends a packet to station Y (from Part D). Station Y will fail to detect the collision and receive the packet. Station A will detect the collision and retransmit. Station Y will receive a second copy of the same packet upon retransmission. Thus Station Y receives duplicates of the same packet.

[Hacker pretends to be V by changing its source address]

2. A) The packet will end up on LAN1, LAN2, and LAN3 as all of the bridge tables are initially empty.
B1 will think that V is on LAN1.
B2 will think that V is on LAN2.

B) The packet will end up at H since B1 thinks that V is located where the Hacker is. The Hacker must be in promiscuous mode to pick up the packet.

C) Say that this series of events occur:
① V sends a packet to S.
- Bridges learn that V is actually below them (on LAN3) instead of on LAN1.

② S sends a reply packet to V.
- Since the bridges have learned that V is on LAN3, the packet reaches the real V instead of H.

③ Upon receiving the reply packet, V will send a RESET packet.
- Communication between H and S stops.

D) To avoid these kinds of attacks, a solution would be to occasionally let packets travel through the bridges (B1 and B2) even if the bridges have already mapped out where they believe the receiver is located. This would allow V to receive some unsolicited packets and hopefully send a RESET packet to S in order to stop the hacker. Doing this is slightly more inefficient, but safer against these attacks.

3. A) One solution to prevent packets from circulating in a loop by having each of the bridges detect if it has already received/seen a particular packet already. To implement this, allow each bridge to have a small cache so it can "remember" the most recent packets it has seen. The bridge can then check if it has already seen any of the incoming packets by referencing its memory. To go even further packets can be marked with a "direction" bit (0 for counter-clockwise, 1 for clockwise) to detect if packets are coming from the other direction.

B) If packets get dropped, then we no longer have Alyssa's guarantee that a packet will circulate both clockwise and anti-clockwise. Alyssa's idea relies on the fact that a packet circulates in both directions so that it can be easily detected as being a looped packet. If a packet from one of the directions gets dropped, it becomes impossible to determine if it is a looped packet - the loop will go undetected and we go back to the original problem.

4 A) D will make an ARP request to get the MAC Address of A. A will receive this request and send its address (all 1's) back to D. D will then try and send its packet to A using that provided address, but will actually end up broadcasting to all stations because it used a destination address of all 1's. Since the packet was only intended for A and not all the stations, all stations besides A will try to forward the packet to A. The stations may or may not have the address for A. The stations that don't have the address for A will need to make an ARP Request to get the address. This causes the whole cycle to repeat again, possibly spiraling out of control until forcefully stopped. If a station does have the address for A, there are two possibilities. If it is the incorrect address, it ends up broadcasting to everyone and causes another cycle of broadcasts. If the station has the correct address, the packet will reach A. Regardless, this entire situation is unlikely to end well.

B) Using a router helps mitigate the problem slightly. Sending a packet from D to A would be handled by the router, as it would be able to grab the address of A. But then D will try to send the packet to A's address (all 1's) and end up broadcasting the packet. In this case, it would only broadcast to the endnodes on its same LAN - not to all endnodes on both LANs. The same broadcast storm effect occurs, but it is confined to only one LAN this time so it's slightly better. If we try to send from A to C (endnodes on the same LAN), the router doesn't need to get involved but the same broadcast storm effect happens on that LAN. Instead of the original scenario where there is an exponential growth e^T (T nodes can transmit T-1 copies of the packet) we have exponential growth e^M (M nodes can transmit M-1 copies of the packet).