Network Systems (201300179), Test 4

April 4, 2014, 15:45–17:15

- This is an open-book exam: you are allowed to use the book by Peterson & Davie and the reader that belongs to this module. Furthermore, use of a dictionary is allowed. Use of a simple (non-graphical) calculator is allowed.
- Other written materials, and laptops, tablets, graphical calculators, mobile phones, etc., are not allowed. ***Please remove any such material and equipment from your desk, now!***
- Although the questions are stated in English, you may answer in English or Dutch, whichever you are more comfortable with.
- You should always explain or motivate your answers, with so much detail that the grader can judge whether you understand the material; so just saying "yes" or giving a formula without explanation is not enough.
- Visiting the toilet without explicit permission of the supervisor is not allowed. During the last 30 minutes of the exam, no toilet visits are allowed.

1. TCP congestion control

Two hosts, A and B, are communicating using a TCP connection. A and B are connected by a router, R, and two links, A-R and R-B. The link A-R has a very high bandwidth, and negligible propagation delay. The link R-B is limited in its data rate, and has a non-negligible propagation delay. A is transferring data to B, and is assumed to have always data in its buffer to transmit. There is no other traffic on the links. Transmission delay for ACK packets from B to A is negligible.

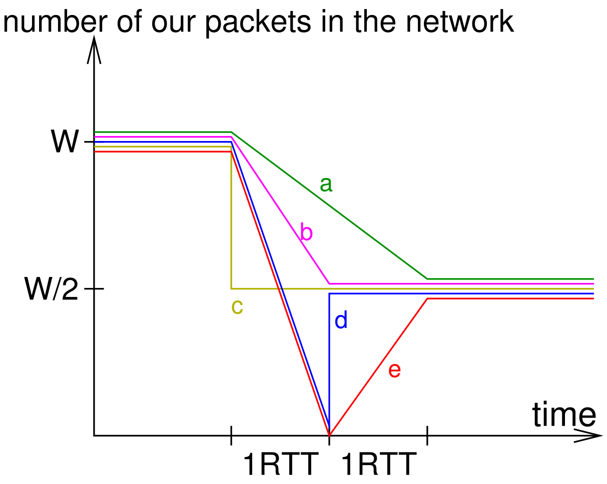
3 pt (a) In which two different ways can host A deduce that a packet was (probably) dropped? For each of the two ways of drop detection, explain how A will adapt its sending rate after the detection, in case of standard (new Reno) TCP.

A can detect loss by means of a time-out. In that case it will reduce its sending rate to 1 MSS per RTT (congestion window of 1 MSS). Alternatively, A can detect loss after receiving 3 duplicate acknowledgements. In that case it will half it sending rate (half the congestion window).

2 pt (b) Suppose that the first data packet sent by A (after the 3-way handshake) is dropped by R. In which of the two ways mentioned by (a) will A detect the loss of this packet? Explain your reasoning.

If the initial packet from A is dropped, no successive packets are being sent to trigger the sending of (3) duplicate acks. So, this packet loss can only be detected by time out.

Please look at the figure below. The figure displays the number of packets in transit from A to B as a function of time. *W* is the size of the congestion window (in packets of size MSS) just before the a (single) packet loss. The fast retransmit / fast recovery procedures in TCP have the macroscopic effect of halving the congestion control window after packet loss. The effect is that also the number of packets in the network is halved, as is indicated in the figure below. How the number of packets in transit evolves during the RTTs directly following the packet loss detection depends on the exact implementation of fast recovery.



3 pt (c) Suppose we would like to follow the purple line (labelled b): the number of packets in transit gradually decreases during 1 RTT, and after 1 RTT, it stabilizes at half of the congestion window before the loss. Design an algorithm that TCP could follow to achieve this, by specifying what actions the sender should take upon receiving acknowledgements in the period from the packet loss detection until the number of packets in transit stabilizes at $W/2$.

From the moment of packet loss (detection), the left side of the congestion window will not proceed until the acknowledgement for the retransmitted packet has been received. In the mean time, packets do leave the network, one for each ack received. In the course of 1 RTT, W packets will leave the network. In order to gradually decrease the number of packets in transit from W to W/2, a new packet must be sent for every two received acks. Since the left side of the congestion will not proceed, this means that the congestion windwo size must be increase by MSS/2 for every received ack. This also applies to the already received 3 duplicate acks, so at the moment of loss detection (3 dupl acks), the congestion window should already be incraesed with 3/2 MSS. At the moment the ack for the retranmitted packet is received, all of a sudden the left side of the congestion window will move with W MSS because the ack for the retransmitted packet will cumulatively acknowledge all W packets that have been received since the loss. To compensate for that, the congestion window size will have to be reset to W/2 at that moment.

2. Quality of Service

The company fly-by-night has an Internet connection that is only just capable of handling the entire company's data traffic. Their network administrator wants to ensure that the company's web servers have good upload speed to the Internet. He manages to install a QoS-enabled router that connects the company's internal networks to the Internet link. This router has functions implemented for packet classification, policing, admission control, and several queueing disciplines. The network administrator wants to use a certain queueing discipline to give the mentioned traffic a better treatment.

3 pt (a) How can the network administrator avoid that web traffic is completely occupying the Internet link, while starving the company's regular traffic?

He could install a scheduler with WFQ, where the weight is such that a significant part of the bandwidth is for the ”high-priority” queue. The remaining capacity is then available for the other traffic. Alternatively, he can use priority queueing, but in that case, he has to install a policer that either shapes the traffic, or diverts excess traffic to the ”low-priority” queue. The policer could use a token bucket mechanism that uses a token generation rate that is below the link data rate.

Let us now look at the issue of QoS in a more general way. Suppose we have N sources, which each send packets satisfying a token bucket specification: their token buckets have rate r and bucket size B. The output of the sources must be transmitted over a link with a rate of S bytes per second ($N \cdot r < S$). For simplicity, we assume in the remainder of this exercise that all packets have a size of 1 byte.

2 pt (b) What is the maximum delay for a packet from any of the N sources with FIFO scheduling?

In the worst case, bursts from all N sources are generated at the same time. If that is the case, the first byte of the burst can be transmitted immediately, since earlier bytes must have already been transmitted, since the token buffer is refilled slower than the link transmits data ($S > N \cdot r$). So, the maximum delay is the delay of the last packet of these bursts, which completes transmission after all $N \cdot B$ bytes have been transmitted, i.e., after $N \cdot B/S$ seconds

3 pt (c) Assume priority scheduling is used, with 2 sources using the high priority queue, and the remaining $N - 2$ sources using the low priority queue. What is the maximum delay for a packet from one of the high priority sources?

In the worst case, both high-priority sources send a burst of size B to the high priority queue at the same time, and also a low-priority packet has just started service. The last packet in the buffer (from one of the two high priority flows will finish transmission after $(2 \cdot B + 1)/S$ seconds.

3. Security

3 pt (a) What are certificates used for, for example in the context of HTTPS? And what kind of attack would be possible if certificates were not used? Explain.

Verifying that a (public) key indeed belongs to the person/site it is used for.

2 pt (b) The fly-by-night network administrator is afraid that people will send packets with ”spoofed” IP addresses from his network (i.e., packets of which the source IP address does not actually belong to this network). In order to block such outgoing packets, the network administrator wants to use a firewall. Should he choose a *stateless* or a *stateful* firewall? Explain.

Stateless is sufficient: just drop all outgoing packets whose source address is wrong.

3 pt (c) Recall that WEP has a weakness which makes it easy to find the key after about 300 000 packets have been intercepted. Now suppose that switching to WPA2 is not an option. Propose a way to make WEP secure, by changing its key very frequently.

In your solution, keep in mind that it should be automated (the user must not have to do the key changing by hand), that all nodes should always know what the current key is, and that packets may occasionally get lost.

Of course, there are many solutions.
To automate the keys, one could derive them from a master key, e.g., hash of masterkey and a sequence number. To make sure nodes know which key to use right now, one could tie this to time (e.g., change exactly every second) or periodically announce the current key sequence number in an unencrypted packet.
But solutions where the last packet of the previous key announces what key to use next, can’t work reliably as packets (e.g., a packet that announces that the key must be changed) can get lost. Also, unencryptedly broadcasting the new key itself (rather than e.g. a sequence number) is not a good idea, since evedroppers then would also learn the new key.

Continued on next page...

4. Time synchronization & localization

2 pt (a) (1) What is the difference between accuracy and precision? (2) RBS is using reference broadcast beacons. Give two methods on how its precision be increased.

(1) Accuracy is the measure how close a measured/calculated value towards is actual value, and precision is a measure how much deviation is between the measured/calculated values.(2) Precision can be increased by adding more beacons and by having a faster duty cycle.

3 pt (b) In APIT there are fixed beacons and blind nodes. (1) How does APIT operate if there is only 1 blind node? (2) Can a position be estimated outside the convex hull of the beacons?

(1) APIT cannot operate with only one blind node, as it estimates its position based upon other blind nodes. What it then can do is mainly relying on distance estimates from the received beacons. (2) APIT operates by making a triangle between the beacons. This implies that a blind node cannot be located outside of the convex hull, but instead will be located on the convex hull.

3 pt (c) Wifi localization on mobile phones using RSSI is not very accurate. Give three mechanisms on how to improve its accuracy.

(1) Using dead reckoning with the accelerometer and/or gyroscope; (2) Using fingerprinting of all available radio signals (wifi, GSM, etc.); (3) Using knowledge of the environment, like available maps, or making use of SLAM techniques to make maps and determine possible positions.

End of this exam.