Network Systems (201600146/201600197), Test 2

March 10, 2017, 13:45–15:15

Answers

1. Congestion Control

(a) Since the packet loss was detected by a timeout, we return to slow start, meaning the `CongestionWindow` will become 1 MSS. The `SlowstartThreshold` will be set to *half* the value of the `CongestionWindow` just before the packet loss was detected: 10/2 = 5 MSS.

(b) Answer D.

(c) Answer B.

(d) Answers D and E.

(e) Answer D.

---

2. QoS

(a) Answer C.

(b) Answer B.

(c) It can be used by assigning a weight of more than 60% to the video data. Then effectively more than 60 Mbit/s of the link's capacity is available for video, which is enough for it to never build up a large queue, regardless of the web traffic.

(d) Answers D, E, G.

As for option D: in the 8 ms interval, the bucket easily fills up completely to 1500 tokens. 1000 of those are then used for a packet, leaving 500 tokens, to which another 500 tokens are added during the 2 ms interval, so there are again 1000 tokens when the next packet is due. After that packet, the bucket is empty, but the next 8 ms are enough to fill it completely again, and so on.

Options E and G are also obeyed by our packet flow; in fact, these options would allow even more packets to be sent, if the source would want to.

The remaining options are not enough: A doesn't even allow any packet of 1000 bytes; B and C can send a 1000 byte packet, but can't accumulate enough tokens to send another one 2 ms later; and F does not get enough tokens for the average rate at which our source is sending (2 times 1000 bytes in every 10 ms interval).

Continued on next page...

3. Security

(a)

	What is hidden?		
	destination IP	destination port	user data
SSH	N	N	Y
SSL	N	N	Y
IPSec in tunnel mode	Y	Y	Y
IPSec in transport mode	N	Y	Y

(b) Answer B.

By comparing the fingerprint to the server's real key fingerprint (which you presumably have gotten in some safe way), you check that the public key you just received from whoever you're talking to, matches the real server's key. Thus, you know that if you encrypt anything with this key, only the intended server can decrypt it.

(Strictly speaking, you're still not sure that you're talking to the real server, because any "man in the middle" could send this same public key — it's public, after all; but it is useless for a man-in-the-middle to send the server's true public key, because he won't be able to decrypt what you're going to encrypt with it, since only the real server has the corresponding private key.)

(c) Answer B.

(d) It can't work. The company's internet link itself is severely overloaded, so doing some filtering in the firewall *after* that link won't solve the problem: the link remains overloaded, dropping lots of packets, and being effectively useless.