Telematics	**Assignment 10**
Freie Universität Berlin	Winter Term 2013/14
Emmanuel Baccelli, Matthias Wählisch, Simon Schmitt	

Note: The referenced websites and publications do not have to be read from begin to the end. You are only required to read enough to answer the questions and to get a general understanding of the topics.

1. **Self-Clocking**

   (a) Explain the self-clocking property of TCP.

   (b) How can the self-clocking be disturbed?

2. **Selective Acknowledgements**

   Read RFC 2018 that specifies the *Selective Acknowledgement* (SACK) option for TCP.

   (a) What problem is addressed by SACKs and how are they used in a TCP connection?

   (b) Give an example for a TCP connection using the SACK option where some segments are lost and explain which values are contained in the SACK options in the TCP headers.

3. **Forward Acknowledgements**

   Have a look at the publication Forward acknowledgement: refining TCP congestion control.

   (a) Discuss the *Forward Acknowledgment* (FACK) congestion control algorithm.

   (b) What problem is addressed by FACK and how is it used in a TCP connection?

4. **Proactive Congestion Control**

   TCP congestion control algorithms as implemented in TCP Vegas or TCP-LP are considered to be proactive in contrast to the common reactive algorithms.

   (a) Discuss the difference of the approaches.

   (b) What does the term *TCP fairness* mean?

5. **Explicit Congestion Control**

   RFC 3168 defines an *Explicit Congestion Control* (ECN) approach for IP and transport layer protocols.

   (a) Explain how ECN works.

   (b) Why is the principle of a layered network architecture violated?

6. **Alternative Transport Layer Protocols**

   Although TCP and UDP are the dominating transport layer protocols, there are alternatives.

   (a) Give examples and name the basic features that differentiate these alternative protocols from TCP and UDP.

   (b) Discuss which problems they try to solve.

7. **RIOT: Completing a simplistic network stack**

   Extend your implementation of the previous RIOT task (ID Addressing, Neighbor Discovery).

   (a) Implement a simple Distance Vector Protocol to allow for routing over multiple hops.

   (b) Indicate the successful transmission of ID packets by acknowledgements.

   (c) Add a simple flow control mechanism to your implementation.

   (d) Take a look at the DES-Virt tool (`https://github.com/des-testbed/desvirt`). Follow the instructions to set up a virtual test environment with 5 nodes connected in a *ring* topology:

   ```
   $ ./topology_creator -s 5 -t ring -n riot_native -e <binary>
   ```

   (e) Test your implementation by sending chat messages between the nodes.

   Note: If you have problems, e.g. setting up the test environment, feel free to ask for help on Piazza.