Number:	6. Assignment
Issued:	25.11.10
Tutorial:	03.12.10
Lecturer:	Prof. Dr. Güneş, Dipl.-Inf. Blywis
Contact:	{guenes, blywis}@inf.fu-berlin.de

Exercise 1, Multiplexing and Multiple Access:

Discuss the terms multiplexing and multiple access and give examples. How do these terms relate to the duplex, half-duplex, or simplex properties of particular network technologies?

Exercise 2, Multiple Access Protocols:

Repeat and discuss the different multiple access protocols introduced in the lecture. Discuss how these approaches can interfere with services of upper layer protocols.

Exercise 3, Scaling of Token-Ring:

Discuss how a Token-Ring network scales with increasing ring size. Which layers of the ISO/OSI reference model are defined by IEEE 802.5?

Exercise 4, Efficiency of Token-Ring:

Discuss how Token-Ring networks perform under full load and in low traffic scenarios.

Exercise 5, Adaptive Tree Walk:

Sixteen stations are connected in a network using the adaptive tree walk protocol. The stations are numbered from 1 to 16. How many slots are required to resolve the contention when all stations with prime numbers as addresses want to sent data?

Exercise 6, Wireshark - Ethernet:

Connect your computer to an IEEE 802.3 or IEEE 802.11 network. Start to capture the transmitted and received packets with Wireshark. Generate some data traffic by using your web browser, a file sharing application, etc. End your capture after when you have captured "enough" packets. Create three *IO Graphs* for the number of Ethernet frames containing

- IPv4 datagrams
- ARP packets
- Other network layer protocols

Besides IPv4 and ARP which other types did you capture?

Exercise 7, Ethernet Frames:

1. Which bits/parts of the Ethernet frames captured in the previous Exercise are actually received by a destination that is
   - in the same local area network
   - in a different local area network that can be reached over the Internet
     (Remark: Ignore tunneling approaches)
2. Have a look at the Ethernet frame on slide 5.53 in your lecture script. What is the *preamble* for and is it contained in the data in your capture file? Is the *Frame Check Sequence* present?
3. Are there any frames with `ethertype` $\leq$ 0x0600 in your capture file?

Exercise 8, Packet Crafting with Scapy:

1. Download and install the command-line network packet crafting and injection utility Scapy.
2. Read the documentation. Create, inject, and capture (with Wireshark, tcpdump, tshark, ...) the following frames:
   - An invalid Ethernet frame
   - A broadcast Ethernet frame, with a source address where the *Organizationally Unique Identifier* (OUI) is set to the value assigned to *Fraunhofer IMS*
   - A frame with some ASCII text as payload
3. For this part you need two hosts connected to the same Ethernet. Send an Ethernet frame from A to B by setting the corresponding addresses of the network cards in the Ethernet header. Set the type to 0xFFF and append the string "Hello World" as payload. Capture the frame on host A and B. Did host B receive the same frame that A has sent or was something modified? What has changed and why?

Note: Scapy is written in Python and the tool of choice for further exercises. Other tools, e.g., *Nemesis* are easier to use but Scapy is more powerful.

Exercise 9, Filtering:

Wireshark supports *libpcap* capture filters as well as its own display filter format. Configure Wireshark as follows:

1. Capture only Ethernet frames with ethertype 0xFFF.
2. Display only Ethernet frames with ethertype 0xFFF (without using a capture filter).

Exercise 10, Ethernet - Capture Effect:

Two hosts (A and B) are connected to an Ethernet and we assume that both have unlimited data to transmit. The frames belonging to A shall be denoted as $A_1, A_2, \ldots, A_n$ and the frames of B as $B_1, B_2, \ldots, B_n$.
A and B try to transmit frames A_1 and B_1 at the same time. The frames collide and both hosts select a random waiting time as specified by the standard (*binary-exponential-backoff*). Station A's waiting time is $W_A = 0$ time slots and station B's is $W_B = 1$ slots. A "wins" the contention for the medium access and retransmits frame A_1 while B waits for A to finish.
Station B sends frame B_1 again after the transmission of A_1 is finished. Unfortunately, frame B_1 collides with A_2 as A is also trying to transmit. In this situation, A either waits $W_A \in \{0, 1\}$ slots, while B has to choose a random waiting time $W_B \in \{0, 1, 2, 3\}$.

1. Calculate the probability that A wins this contention after the second collision.
2. Assume that station A wins the second contention and frame A_2 is transmitted while B_1 still has to wait. Calculate the probability for A also winning the next (third) contention.
3. Specify a formula to calculate the probability that A wins the i-th contention. Calculate the probability of A winning the i-th contention with $i \in [1..MAX]$.
4. The term **Capture Effect** denotes a situation in which A wins all contentions after the first collision. Host B cannot send data until the maximum number of retransmissions has been reached and the current frame is dropped. Give the probability of this event.
5. Is the capture effect still relevant in today's Ethernet based networks?