V 1.2

Universidad de León

Bachelor Degree on Computer Science and Engineering

Course on Distributed Systems

Homework no. 2

Published on: 7th - November- 2020
Submission date: 17th-May-2020
Submit via: Agora virtual campus homework submission
Format: Only **pdf** format is accepted; please, include your name and ID in the document.

--- Study Guide ---

1. At all times, have the textbooks by Kindberg *et al.* and Peterson & Davie at hand as well as the presentations, notes, practice scripts and solved exercises in `paloalto.unileon.es/ds`.

2. Make sure that you richly explain the answers that you provide. The explanations that *we* compose have the greatest value for *our* advancement.

3. Properly cite whatever references you consult in completing this homework submission.

Exercises

1. **Introduction to Distributed Systems.**

    a. What notable Distributed System transparency applies to the TCP protocol?

    b. List the most important DS transparencies

2. **OSPF Routing Protocol.** The internetwork diagram from Fig. 1 represents a routing domain based on the OSPF protocol (The circles represent IP routers)

a. Apply the Dijkstra's algorithm at node F to obtain its Shortest Path Tree

b. Obtain node F's FIB (Routing Table).

c. Depict the resulting Shortest Path Tree of node F

d. If possible, depict the Shortest Path Tree of node A based on your previous results

e. Will the contents of node A's FIB be the same as those of node F?

f. Explain the information contained in the LSP sent by node F.

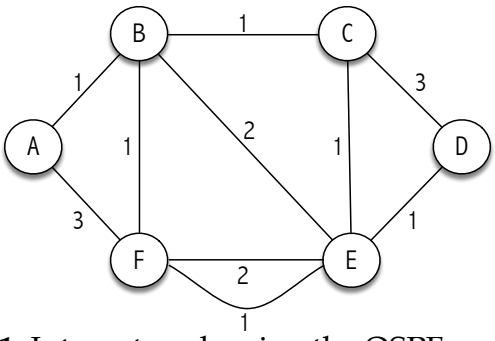

Figure 1. Internetwork using the OSPF routing protocol

3. UDP Protocol.

a. Describe the essential functions of this protocol

b. What are the components of UDP's multiplexing key.

c. Can applications be written that use the UDP protocol for communication over the internet? Explain how.

4. TCP Protocol (i)

a. Describe the essential functions of this protocol

b. What are the components of TCP's multiplexing key.

c. Develop a real, illustrating example of the TCP multiplexing key in your Linux system (Or Windows or Unix) with the help of the `netstat` command.

d. Develop a TCP client that connects with a server that was started by the *system admin* in paloalto.unileon.es and that listens on TCP port 60000. The server was written in C and it uses POSIX Threads for providing concurrent service to

multiple clients. Can *your* client be written in Java or it must be written in C, like the server? I suggest that you extend the TCP client that we used in the first practice about TCP that we did this academic year.

e. After the *system admin* at paloalto.unileon.es started the aforementioned server program, the following Linux netstat command produced the string captured in the ensuing screen dump. Interpret all of its fields.

f. Connect with the server by using your client program such that it sends the following *literal* string:

```
"Send the date"
```

Are you receiving the expected results, *i.e.,* the date string from paloalto.unileon.es?

g. Capture all the packets exchanged by your client and the server from the connection request through the connection close. Again, send the date request string from section f. Use the tcpdump sniffer. Explain the results. We are particularly interested in your explaining the SN (Sequence Numbers) corresponding to the sending of the "Send the date" string and the received ACK SN (Acknowledgement Sequence Numbers).

h. Print out the information about the client socket as the client connection with the server progresses; highlight all the states the client socket goes through.

5. TCP Protocol (ii).

a. Provide an interpretation about the Wireshark trace in Fig. 2 which represents a client trying to connect with a server.

b. What are the relevant TCP multiplexing keys at the client and at the server in the preceding section?

c. Can you identify some exponential backoff distribution in the trace?

d. What's the semantics of the Timestamps that appear on the trace?

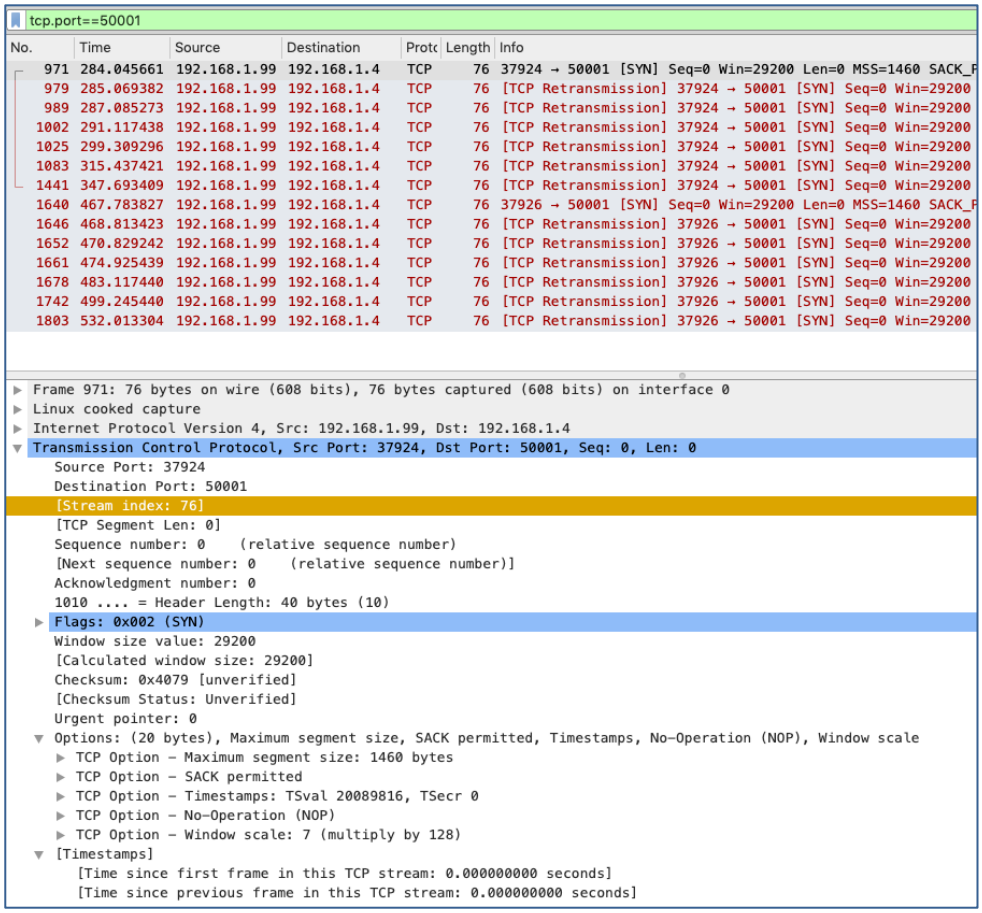

Figure 2. Wireshark trace form a client/server 3-way handshake

6. Clock synchronization exercises:

   a. Do the calculations necessary for synchronizing the clock of host A to that of host B, assume that the minimum Rtt is 10 ms.

   b. Compute the maximum absolute error attained in the synchronization (Consult the presentation slides used for the lectures about Physical Time)

   c. Explain the overall process of synchronization, *i.e.,* using your own words

Synchronization of A's clock

	Timestamps at host A		Timestamps at host B				
Request	Originate TS	TimeOfDay A	Receive TS	Transmit TS	Rtt (ms)	Residence time (ms)	RttN (ms)
1	12:10:32.123	12:10:32.155	12:00:32.160	12:00:32.166			
2	12:10:34.010	12:10:34.045	12:00:34.050	12:00:34.056			
3	12:10:37.484	12:10:37.516	12:00:37.496	12:00:37.510			
4	12:10:40.305	12:10:40.340	12:00:40.329	12:00:40.337			
5	12:10:44.744	12:10:44.772	12:00:44.760	12:00:44.766			
6	12:10:49.015	12:10:49.040	12:00:49.028	12:00:49.041			
7	12:10:55.263	12:10:55.284	12:00:55.285	12:00:55.288			
8	12:10:58.648	12:10:58.669	12:00:58.665	12:00:58.667			
9	12:11:02.340	12:11:02.372	12:01:02.350	12:01:02.357			
10	12:11:08.900	12:11:08.936	12:01:09.000	12:01:09.007			

Figure 3. Synchronization of the clock of host A to the clock of host B