

Network Applications Programming - Homework 4

(UDP-based FTP service)

Motivation:

TCP provides many charming features such as transmission reliability, so FTP service usually adopts TCP. In this homework, you are asked to write FTP server and client, using UDP to “simulate” the reliability behavior of TCP.

Requirements:

In this homework, you will use the UDP `sendto()` and `recvfrom()` system calls to transfer a file from a UDP client to a UDP server. You will need to develop some mechanisms to implement a reliable file transfer because some datagrams sent by your client will be intentionally dropped on the network. To create the datagram dropping environment, when your server receives a UDP datagram, you server can intentionally drop it (acts as if the datagram were not received at all) based on a probability using the `random()` procedure. You have to do fragmentation before sending the file. The size of each fragment is set to 1KB (1024 bytes).

To generate a probability of 0.1 of dropping a datagram, you can include the following piece of code in your program:

```
p = (random()%100)/100;  
if (p < 0.1)  
    drop the datagram;
```

Your server should be aware of such dropping **ONLY** when it receives the “next” datagram from the client. In other words, although the server intentionally drops a datagram, it pretends not to know such a behavior. Thus, in the homework you need to implement an acknowledgement scheme, following TCP’s behavior (i.e., cumulative ACK), to support the transmission reliability. Your server and clients should show some necessary messages to indicate the behaviors of datagram loss and retransmission, so that TAs can be convinced that your program indeed adopts a datagram dropping mechanism.

[Bonus]

You can get extra bonus if you simulate the out-of-sequence transmission scenario and use the ACK mechanism to deal with this scenario. In this case, you may need to reorder the transmission of some fragments. The occurring probability of this scenario is set to 0.05 to observe its effect. Of course, you should write down your observation in the report accordingly.

Grading Policy:

You should prepare a report to tell TAs how you design a reliable file transfer and what interesting things you have observed when running your program (e.g., TCP behaviors). You need to submit your codes and demonstrate your program to TAs. The due day is **5/25**. You will get no point in this homework if you do NOT demonstrate the program (even if you submit your code). Discussion among your classmates is encouraged. However, plagiarists will get ZERO point. The report should be handed in to TAs when demonstrating your program. Below is the grading policy of the homework:

Items	Points
The correctness and performance of your program	50%
The user interface (i.e., messages to show all behaviors)	10%
Code comments & README file	10%
Bonus: Out-of-sequence scenario	50%
Report	30%