Assignment 1 - Graduate Operating Systems - Fall 2021

All of the course projects for this class will involve measurement of some kind. This project will help you to get thinking about how to measure and evaluate a computer system.

For this assignment, you will be measuring a set of actions and configurations using the system clock. Call gettimeofday or other appropriate functions before and after the measurement points and subtract to get the elapsed time. Of course, it is never enough to measure a value once: the measurement may be inherently variable, or the measuring equipment may have its own internal variation. For this work, you should repeat each test about ten times, and report the average and standard deviation.

Setup

Write two programs (one client, one server) that use ZMQ (Zero Message Queuing - https://zeromq.org/) to send a file from one program to the other. For this setup, you can use any languages of your choosing as ZMQ is available across a wide variety of languages. You will be using the Request-Reply model which means that for every request, there will be a reply, e.g. every message will have a response.

Your client code should take in three arguments:

```
% python sendviaZMQ.py FileToSend ReadSize WriteSize
```

The parameters represent the following:

- FileToSend: The file to send / transmit
- ReadSize: The size (in bytes) to read at a time from the file
- WriteSize: The size (in bytes) to write at a time across the ZMQ socket

Your server code that receives the file from the client should take in a two arguments:

```
% python receiveviaZMQ.py FileToWrite WriteSize
```

- FileToWrite: The name of the file to write to
- WriteSize: The maximum size of any writes

For your socket, you can use the following network parameters:

- Transport: TCP or UDP – either is fine
- Host Name: localhost
- Port Number: Something reasonably high with a bit of randomization (e.g. 42879). It is only necessary to pick a port for the server. It is OK to pick a hard value in your code.
    - `socket.connect("tcp://localhost:42879")`

Measurement

With your code working, conduct the following measurements for both the client and the server side:

- Pick three file sizes: A small file size, a medium file size, and a large file size
- Vary the read and write sizes across a reasonable range (e.g. 1000 bytes to 20000 bytes – step size of 1000).
- Pick an appropriate vantage point to measure / record the time that is reasonably repeatable

Present and discuss the results that you observe across the various file sizes as you vary the read and write sizes.

- How consistent are the results across each run?
- Do the results plateau in terms of performance as you change either the read or write times?
- Which has more of an impact, the read size or the write size?

Submission

Submit your assignment individually (this is the only one) via Sakai. Include your code as either a tar / gzip or zip file. Include a document that graphs your various results as a PDF.