Matthew Koken

COEN146 Lab 1

In order to test the transfer time of a file, a file of a large size is needed. Once created, we can measure the time for transfers of different file sizes given the below script:

time cat test | head -c <file size> | ssh [mkoken@linux.scudc.scu.edu](mailto:mkoken@linux.scudc.scu.edu) “cd /home/mkoken && cat -> test1”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| File Size (KB) | File size | Real Time (s) | User Time (s) | Sys Time (s) | Total CPU Time (s) |
| 1 | 1K | 0.012 | 0.005 | 0.007 | 0.01 |
| 10 | 10K | 0.175 | 0.014 | 0.018 | 0.03 |
| 100 | 100K | 0.164 | 0.017 | 0.016 | 0.03 |
| 1000 | 1M | 0.188 | 0.025 | 0.022 | 0.05 |
| 1024 | 1024K | 0.738 | 0.023 | 0.018 | 0.04 |
| 10000 | 10M | 0.360 | 0.052 | 0.040 | 0.09 |
| 100000 | 100M | 2.040 | 0.284 | 0.167 | 0.45 |
| 1000000 | 1G | 24.129 | 2.990 | 1.857 | 4.85 |
| 1024000 | 1024M | 21.290 | 2.947 | 1.643 | 4.59 |

In order to measure the effective bandwidth, we look at the total time the transfer script spent in the CPU. This includes both user and system time. There will be a baseline, or minimum time for any file transfer. The baseline time will be the time needed to establish the connection and any other overhead required to be able to transfer a file. This can be measured in the same manner as the other file transfers by providing a 0 file size. The effective bandwidth then will be the total amount of data able to be transferred per unit of time. The total transfer time will be the total CPU time spent on the transfer operation – establishing the connection and then completing the transfer of data. The bandwidth, then, would be Total CPU Time / File Size.

Given a reliable network, it would be reasonable to expect a fixed amount of time to transfer data. Therefore, the total CPU time should appear as a linear increase based on the size of the file to be transferred. Given the first 5 data points, this trend appears as expected. However, the graph seems to increase exponentially for the additional data points. This appearance is incorrect – The graph does not properly account for the rapidly increasing file size. If more data points had been taken at smaller intervals, the graph would take on a much more linear appearance. Looking at the data points for the 1000 and 1024 KB file transfers, the total CPU time is very close. This makes sense because the file sizes are very similar. But for the 100000 and 10000000 KB transfers, there is a much larger gap between the two file sizes, and a corresponding large jump in total CPU time. If the graph were to be expanded, the linear progression would be more apparent.

From this experiment, we see how CPU time is related to the time it takes to transfer a file over a network, or bandwidth. Given a transfer with a file of size 0, we can establish a baseline for the transfer – the time it takes to make the connection.