Department of Electrical Engineering
The Cooper Union for the Advancement of Science & Art

System Calls and Error Reporting

by

Benjamin Schenker

1 Introduction

A simple C program was written to utilize UNIX system calls for file I/O and properly handle and report error conditions. A script was then written in MATLAB to measure the throughput of the program for various buffer sizes.
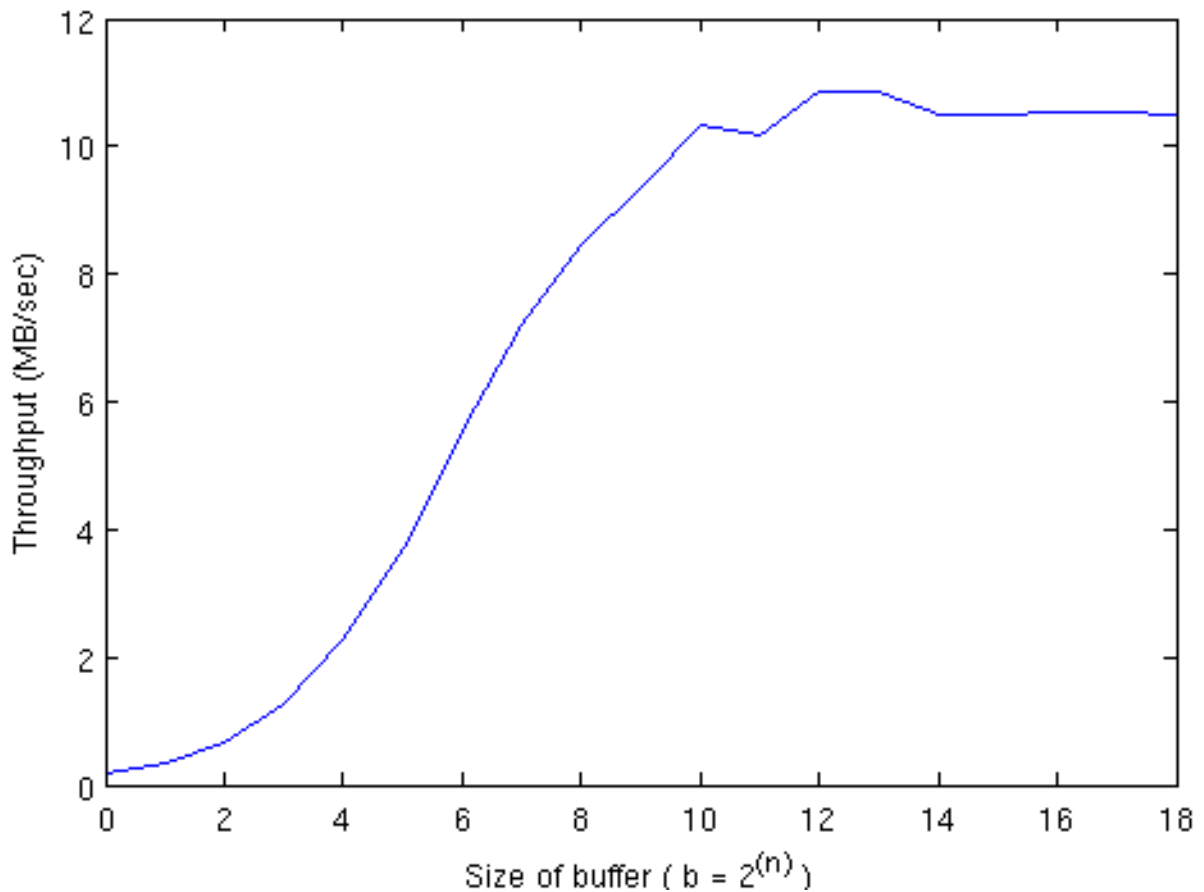
2 Results

Figure 1: Throughput of copycat with varying buffer size

The script executed the program taking input from a static 2.2MB file of random bytes (previously drawn from /dev/urandom) and copying it to a separate file. The only change between calls was

the specified buffer size. As is shown in figure 1, the throughput is quite low for very small buffers but there is only minor gain once the buffer is larger than 1024 bytes.

3 Analysis

There are many reasons that the buffer size would affect the throughput in this way. One major contributor is that a smaller buffer requires more system calls, which can be quite time consuming. For example when the buffer size is 1 byte, the program has to make 4.4 million system calls (a read and a write for each byte). This explains the very noticable 12 second runtime for the program using a 1 byte buffer. This effect is mitigated as the buffer size increases such that the frequency of the system calls is closer to that of the normal timer interrupt.

4 Testing Environment

The tests were performed on a Dell Vostro with an intel Core i7 processor running ArchLinux (in the SeniorLab). The OS uses AFS and ACLs instead of UNIX permissions. While it ignores group and world UNIX permissions AFS does utilize the user level UNIX permissions. Thus user level permissions were modified to test the error handling of the program.

5 Conclusion

The project was successfuly completed and the effect of the buffer size on the throughtput was measured and discussed.