CS105 – Computer Systems Spring 2020

Problem Session 5: File and Network IO

SOLUTION

Wednesday, April 29, 2020

1. Consider the following code:

```
int main(int argc, char* argv[]){
    char buf[3] = "ab";
    int r = open("file.txt", O_RDONLY);
    int r1 = dup(r); // equivalent to dup2(r, r1);

    read(r, buf, 1);

    int pid;
    if((pid=fork()) == 0) {
        r1 = open("file.txt", O_RDONLY);
    } else{
        waitpid(pid, NULL, 0);
    }

    read(r1, buf+1, 1);

    printf("%s", buf);

    return 0;
}
```

   Assume that the disk file `file.txt` contains the string of characters `CS105` . Also assume that all system calls succeed. What will be the output when this code is compiled and run?

   This code will print the string `CCCS`.

2. Logging daemons are background processes whose purpose is simply to accept messages from clients and log them to a file. (This is sometimes used in secure networks to ensure that, even if an attacker breaks into a computer, he cant erase his traces from the logs because a copy of the log is stored elsewhere.)

   In the following table, please list the UNIX I/O function calls that each side of the connection would make. Each row should contain the name of a single system or library call, placed in the appropriate column. Please list the calls in the order they are called, not the order they return. Please only include network-related operations (accept, bind, close, connect, listen, read, socket, and write).

   You should make the following assumptions:

   - The server and client are communicating over TCP.
   - The server finishes initializing before the client starts.
   - The server only serves a single client.
   - The client already knows the servers IP address.
   - The client only sends one message before closing the connection.

Client	Server
	`socket`
	`bind`
	`listen`
	`accept`
`socket`	
`connect`	
	`read`
`write`	
`close`	