CMPE 279 - Assignment 3

Ke He 012561380, Qiyue Zhang 012550096

1. What happens if the file being requested is not inside the chroot? Does it matter?

It is not a problem in this case because the disk file is already open before fork/exec and we've started the program with root privilege, thus the program is still under root privilege and it has access to any readable files.

2. Explain the mechanism you are using to pass file descriptors between parent and child processes.

File descriptor need to be passed from parent to child process (the one after exec). The child process after fork still have access to these variables so no need to consider about it.

I use command line argument to pass the file descriptor. When I enter the re-exec process, I check if the argument contains "-d", if it doesn't, that means there is no file descriptor passed in; if it contains "-d", I passed the argument after "d" as file descriptor.

3. What happens if the file size of the disk file exceeds the size of the client's hardcoded buffer? Does the client have sufficient checks to ensure no buffer overruns occur?

The client is set to reads 1024 bits of data, it is set in client's hardcoded buffer size, so buffer will only try to read and store first 1024 bit, others will be ignored.

It is not sufficient for the current approach, since the input is from outside, it going to be unpredictable, we cannot always guarantee the length, the best approach would be dynamically allocating the memory according to the input size. In this scenario, the program can ensure to read all data and buffer size would not be overflowed.