Could you write a command line shell using threads rather than processes? Why or why not? What would be the advantages and disadvantages of this approach?

Yes, you can. Because all you are doing is changing the input into an thread rather than a process. A thread still can do everything that we want it to do and still have the ability to share data between threads. Even if ran the threads concurrently, It will still feel like it’s nothing has changed by the end user perspective.

Some advantages of using a threaded solution is that it would be cheaper to use a thread. Process creation is time consuming and resource intensive. We could say it would be more responsive. I think the application in nature is not something, in my opinion, that needs to be super responsive. You can only type things to it. Another benefit that would be useful is the way the resource is being shared. The threads are all in the same process and they are not going to different a share pool in memory. Finally, it can be scalable. For whatever reason the terminal program starts doing complex things, the program can always make more threads and even run parallel in different cores.

How do you resolve name collisions between internal shell and system programs? For example, if there is a system program called “alias” – how would you execute it?

As a programmer, it would be easier to just set the shell’s alias command to, “set alias” and “remove alias” just to make it easier to code. If we know this is going to run in a Linux, or mac environment and the alias command is always known as a system call.

If the shell needs its own alias command, I would make the program check the bin folder or whatever path is chosen, then if the command ‘alias’ comes up, make the program switch its own alias to something else such as, “set alias” or whatever would sound appropriate.

So using my program as an example (snipped allot of code out)



How would you run a shell under a different user code? What would you have to require to order to do it? Would it be different just to run a command? (like sudo)

With root turned on you can now do things you wouldn’t be able to do with normal access privileges. The same is with sudo. In terms of programming one. If I know the user is going to sudo allot or will be running in root. I would put, “are you sure prompt” for the more lethal commands. Also, we can do a rollback command if for less lethal stuff such as removing the bin or something in that nature.