Address Space of a Process

January 31, 2018

All of the child process were printing the same local and global address. This happens due to virtual addresses. When we execute ./a.out, each child process is assigned the same virtual address. Each process thinks that it has the entire machine to itself. So parent process' pages are shared among them and not copied. Linux uses a copy on write (COW) mechanism. With this technique, when a fork occurs, parent process' pages are not copied for the child process, instead they are shared between child and parent process. Whenever a process wants to modify a page, a seperate copy of that page alone is made for that process. The process will then use newly copied page for future references rather than the shared one. This is called copy on write since the page is copied when one process writes to it.

Global address is same everywhere because they are not shared in any way which is visible to the programmer. The process can modify their own copy of global variable independently and they can change without any noticeable effect on the other process.