同济大学计算机网络

实验报告



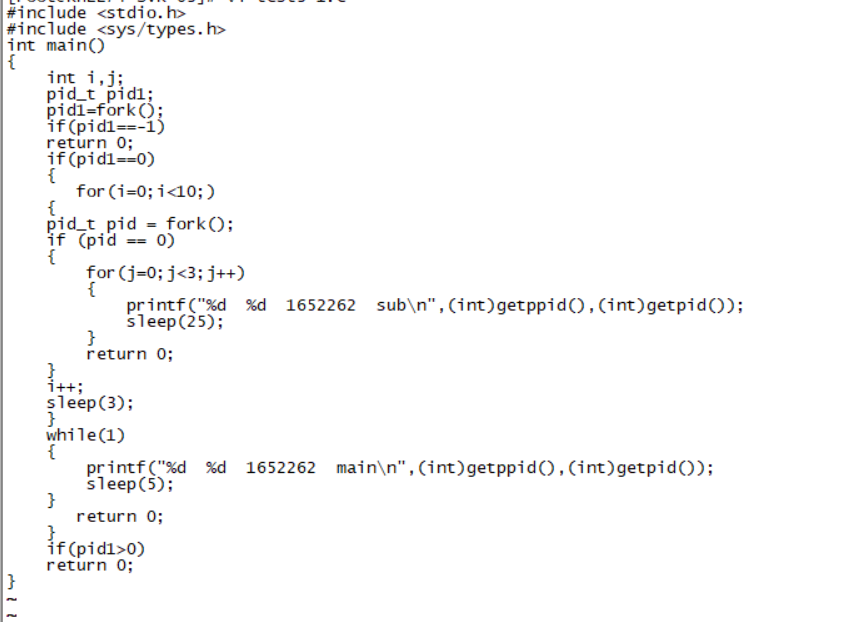
姓名： 涂远鹏-1652262

三.守护进程再次分裂子进程

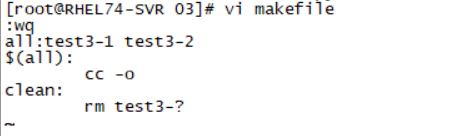
1.在03子目录下写test3-1.c，循环10次，每隔三秒产生一个子进程........写配套的makefile文件，make后可生成test3-1可执行文件：

答：

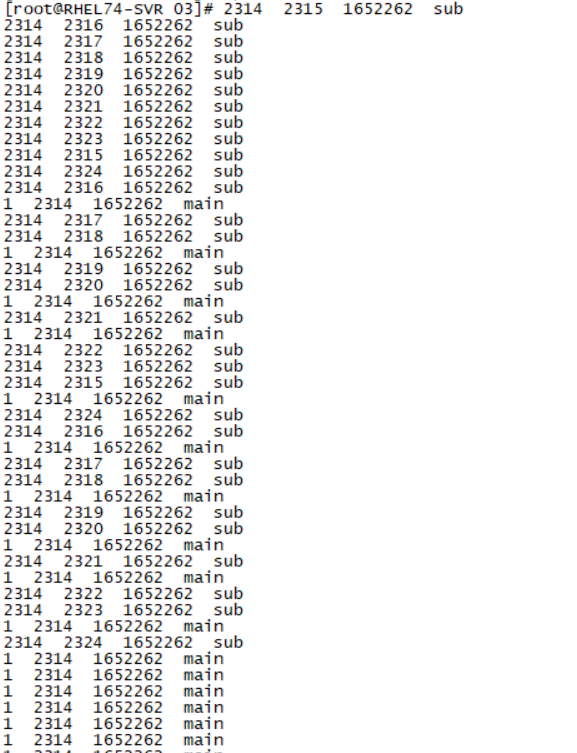
编写test3-1.c：



编写makefile：

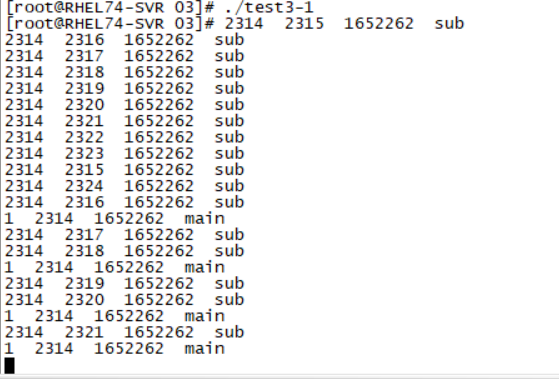


生成的可执行文件测试结果：



2.分裂出的子进程和守护进程之间的进程id有何关联？

答：分裂出的子进程的id均比其父进程(守护进程)的id要大并且是从守护进程的id开始进行+1增长的：



3.什么叫僵尸进程？僵尸进程的产生原因？

答：

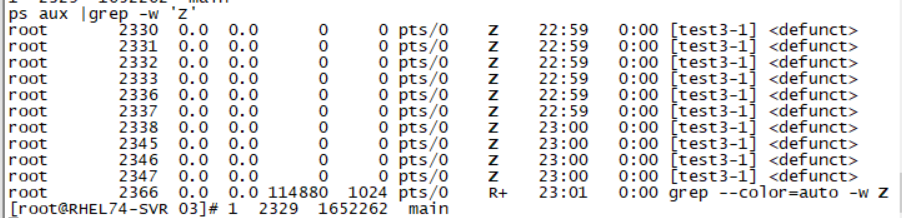
僵尸进程：

一个子进程在其父进程还没有调用wait()或waitpid()的情况下退出。这个子进程就是僵尸进程

产生僵尸进程的原因：

一个进程在调用exit命令结束自己的生命的时候，其实它并没有真正的被销毁，而是留下一个称为僵尸进程（Zombie）的数据结构（系统调用exit，它的作用是使进程退出，但也仅仅限于将一个正常的进程变成一个僵尸进程，并不能将其完全销毁）。在Linux进程的状态中，僵尸进程是非常特殊的一种，它已经放弃了几乎所有内存空间，没有任何可执行代码，也不能被调度，仅仅在进程列表中保留一个位置，记载该进程的退出状态等信息供其他进程收集，除此之外，僵尸进程不再占有任何内存空间。它需要它的父进程来为它收尸，如果他的父进程没安装SIGCHLD信号处理函数调用wait或waitpid（）等待子进程结束，又没有显式忽略该信号，那么它就一直保持僵尸状态，如果这时父进程结束了，那么[init进程](https://baike.baidu.com/item/init%E8%BF%9B%E7%A8%8B" \t "https://baike.baidu.com/item/%E5%83%B5%E5%B0%B8%E8%BF%9B%E7%A8%8B/_blank)自动会接手这个子进程，为它收尸，它还是能被清除的。但是如果父进程是一个循环，不会结束，那么子进程就会一直保持僵尸状态，这就是为什么系统中有时会有很多的僵尸进程。

查看test3-1生成的僵尸进程(ps aux |grep -w ‘Z’)：

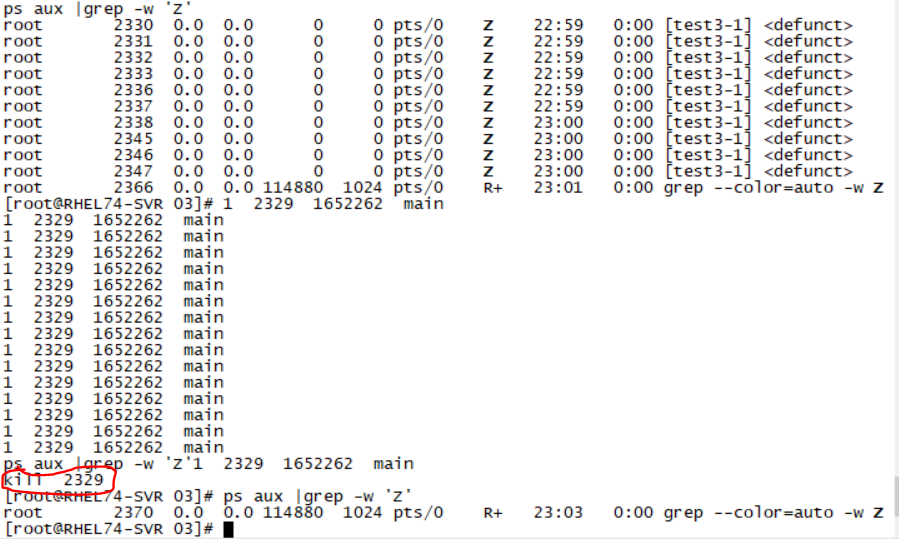


4.如何杀死僵尸进程？

答：

使用Kill -HUP 僵尸进程ID来杀死僵尸进程，往往此种情况无法杀死僵尸进程，此时就需要杀死僵尸进程的父进程，僵尸进程则被init进程，处理kill -HUP 僵尸进程父ID：

通过杀死其父进程的方法杀死其子进程：



5.写test3-2.c，要求同test3-1，但是子进程退出后不能存在僵尸进程：

答：

子进程在终止时会向父进程发SIGCHLD信号,Linux默认是忽略该信号的，我们可以显式安装该信号，在信号处理函数中调用 wait 等函数来为其收尸,从而实现要求

编写的test3-2.c如下：



暂停查看僵尸进程发现并没有产生僵尸进程，测试结果，并未产生任何僵尸进程：

