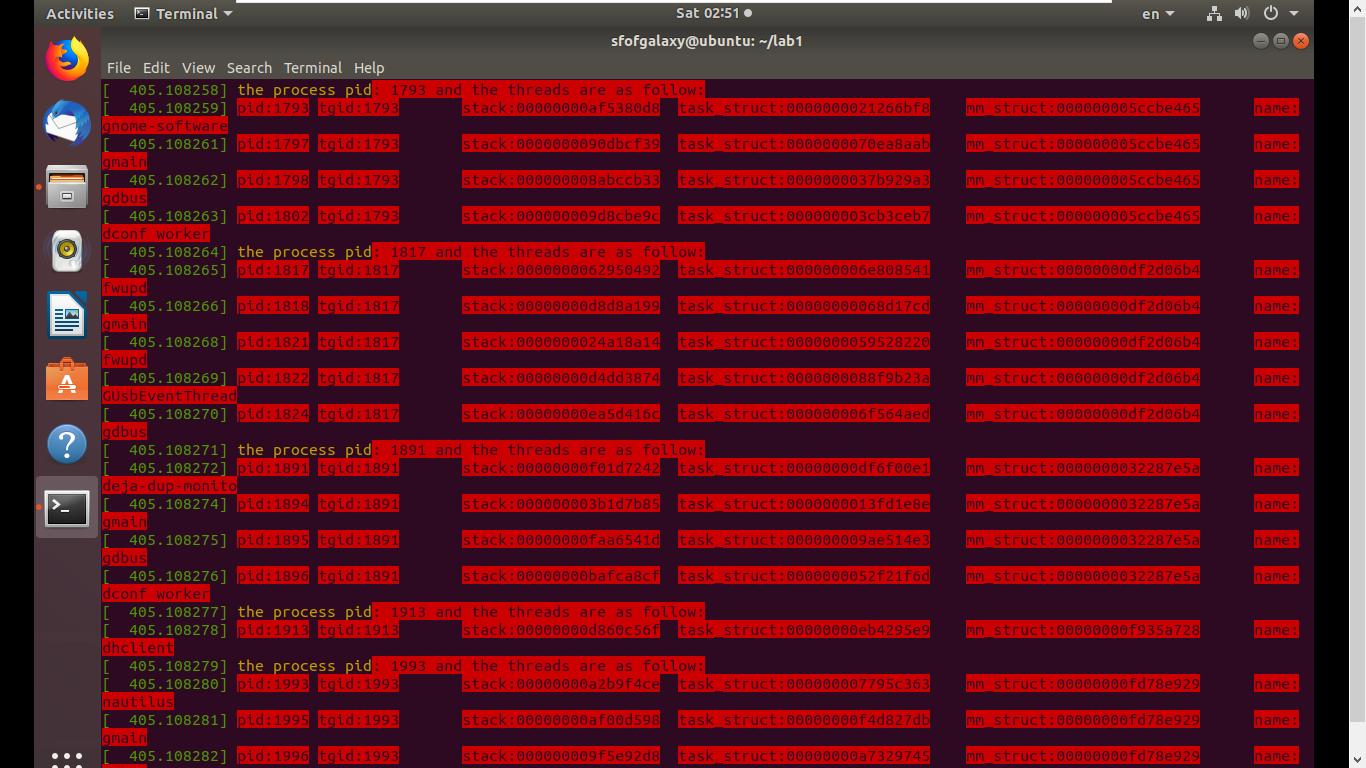
OS LAB1 report

## 加载模块，输出显示进程以及线程信息，截图如下：







代码和本报告一起上交

其中先输出进程号，在输出进程下的所有线程group\_thread和进程是一个task\_struct

## 如何区分进程和线程，根据什么判断

根据pid和tgid 如果 pid和tgid相同，则为进程，不相同则为tgid进程下的线程组号下的线程。

## 我们为什么要使用内核模块，内核模块有何优缺点？

一、优点:

1. 使得内核更加紧凑灵活。
2. 修改内核时, 不必重新编译整个内核。
3. 模块可以不依赖于某个固定的硬件平台,即使它依赖于某系统特殊的硬件特点
4. 模块的目标代码一旦被链接到内核,它的作用和静态链接的内核目标代码完全等价。
5. 所以当调用模块函数时,无须显式的消息传递。
6. 截获系统调用。

二、 缺点:模块机制带来安全上的隐患

1. 修改输出符号表。
2. 通过修改系统调用,使模块自己隐性。
3. 重定向文件操作。
4. 隐藏进程。
5. 重定向可执行文件。
6. 通过修改系统调用sys\_socketcall,使得系统在收到指定数据包后打开后门

## 线程和子进程有何区别？

相同点：

1. 二者都具有ID，一组寄存器，状态，优先级以及所要遵循的调度策略。
2. 每个进程都有一个进程控制块，线程也拥有一个线程控制块。
3. 线程和子进程共享父进程中的资源；线程和子进程独立于它们的父进程，竞争使用处理器资源；线程和子进程的创建者可以在线程和子进程上实行某些控制，比如，创建者可以取消、挂起、继续和修改线程和子进程的优先级；线程和子进程可以改变其属性并创建新的资源。

区别：

1. 线程是进程的一部分, 一个没有线程的进程是可以被看作单线程的，如果一个进程内拥有多个进程，进程的执行过程不是一条线（线程）的，而是多条线（线程）共同完成的。
2. 启动一个线程所花费的空间远远小于启动一个进程所花费的空间，而且，线程间彼此切换所需的时间也远远小于进程间切换所需要的时间。
3. 系统在运行的时候会为每个进程分配不同的内存区域，但是不会为线程分配内存（线程所使用的资源是它所属的进程的资源），线程组只能共享资源。
4. 进程是系统所有资源分配时候的一个基本单位，拥有一个完整的虚拟空间地址，并不依赖线程而独立存在。

## 说明list\_head结构的作用。

list\_head作用主要是连接不同进程和不同线程，双向链表进行遍历，便于添加和删除进程和线程。

进程则为进程直接的相互连接与遍历，线程则为该进程下的线程的遍历。