基于你之前做的工作，我们来进一步分析一下内核。请你在现有内核的基础上，找到五六个配置项（找这些配置项：它们要么新定义了一些全局变量，要么在已有的全局结构体类型里定义了几个新成员），将它们设置成y，使它们管理的内核代码被编到内核里，继续获取CFG，然后可以跟之前的CFG比较一下，确认以下三个问题：

1. 浏览内核源码，看这个配置项新增了多少行代码？

2、这个配置项在CFG里新增了多少代码块？占整个内核CFG块数的比例是多少？

3、新增的源码里是否有对这些新变量/新成员进行读写的语句？如果有，它们占新增的总代码行（刨除类型定义语句）的比例是多少？包含读写语句的代码块占新增代码块的比又是多少？

usb\_audio\_midi\_driver

general setup->page allocator randomization 不涉及全局变量或新全局结构体类型里的新成员

general setup->memory placement aware numa scheduler

general setup->Checkpoint/restore support

第一个比较简单，

1623 export WLLVM\_BC\_STORE=/home/ubuntu/code/bcfiles

1624 export LLVM\_COMPILER=clang

1625 make CC=wllvm defconfig

1626 make CC=wllvm LLVM=1 -j8