**基于系统调用重载的文件访问日志原型系统说明**

该原型系统由运行在应用层的日志应用程序和执行在内核层的内核日志模块两部分组成，这两部分的编译方法不同，需要分别进行编译。

**1、应用程序的编译**

应用程序的编译只需在shell命令行输入下述命令即可：

gcc –o auditdemo auditdemo.c

其中auditdemo.c是保存控制配置程序源代码的文件名，假定该文件就在当前目录下，auditdemo为编译出的目标文件，该目标文件名可以在gcc编译时任意命名。gcc命令执行结束，就会在当前目录下得到一个auditdemo的可执行文件。

**2、内核模块的编译**

本开发实践通过make工具来完成内核模块的编译，具体的Makefile文件内容为：

obj-m:=AuditModule.o

AuditModule-objs :=sdthook.o syscalltable.o netlinkp.o

KDIR := /lib/modules/$(shell uname -r)/build

PWD := $(shell pwd)

$(MAKE) -C $(KDIR) SUBDIRS=$(PWD) modules

赋值语句Obj-m = AuditModule.o说明要使用目标文件AuditModule.o建立一个模块，该模块的名字就是AuditModule.ko。赋值语句AuditModule-objs:=sdthook.o syscalltable.o netlinkp.o指明了AuditModule.o所依赖的目标文件，这些目标文件将由同名的\*.c文件编译而成。

在当前目录（即包含上述的Makefile文件和源代码文件的目录）下输入 make，执行完成后，就会看到AuditModule.ko文件，这就是可加载的内核模块目标文件。

需注意的是，在不同内核版本下，获取系统调用表的方式可能不一样，Netlink函数接口也有可能不一样，需依据特定的内核版本进行调整。