用户从用户态切换到内核的方式有两种：

1.被动式：即，没有主动发起请求，包括硬件中断和程序异常。

2.主动式：用户态程序发起命令请求，ARM响应进入特权模式而Linux切入内核态，即，系统调用。

被动式原理：当硬件中断到来时，通过硬件直接给CPSR置位、ARM进入IRQ模式，Linux系统进入内核态，或者用户态程序出现异常、硬件自动完成上述工作。

主动式原理：软中断，Linux内核给用户空间开放一个可以直接操作硬件寄存器进而引发中断的机制-----系统调用。当用户态调用系统调用函数时，相应的硬件被置位引发中断，导致ARM工作模式的切换，进入Linux内核态。

系统调用getuid（）的定义：

#include <unistd.h>

int getuid(void){

long \_\_res;

\_\_asm\_\_ volatile(“SWI”) //x86上是init $0x80

: “=a” (\_\_res)

: “” (\_\_\_NR\_getuid);

\_\_syscall\_return(int,\_\_res):

}

Linux通过SWI指令实现系统调用，SWI将使用中断/异常向量号128将控制权移给内核。

SWI指令ARM从用户模式进入管理模式，即，Linux操作系统从用户态进入内核态。