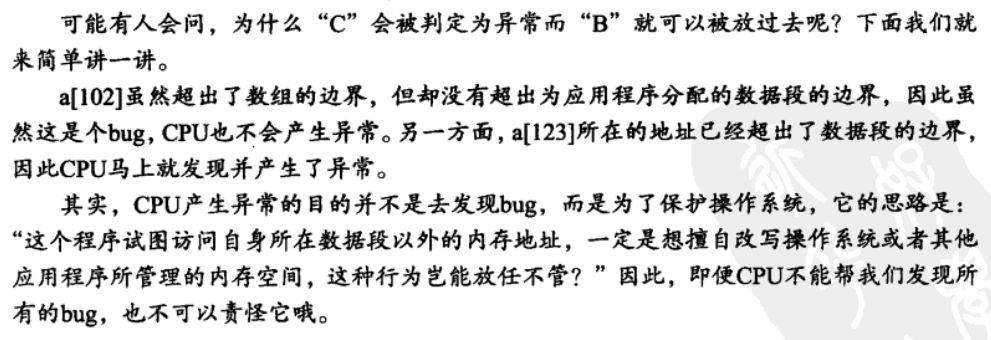
删除crack，编写bug1.c，增加栈异常中断,IDT设置中添加，修改Makefile

记得在.h中声明

QEMU中对异常模拟有问题，在VirtualBox中测试





编写无限循环的bug2和不断输出a的bug3，修改Makefile

修改0c和0d

修改Main中按键判断，使用 左Ctrl+c结束程序

 int key\_Ctrl = 0;

    struct CONSOLE \*cons;

                if (i == 256 + 0x1D)

                { /\*左Ctrl ON \*/

                    key\_Ctrl |= 1;

                }

                if (i == 256 + 0x9D)

                { /\*左Ctrl OFF \*/

                    key\_Ctrl &= ~1;

                }

                if (i == 256 + 0x2E && key\_Ctrl != 0 && task\_cons->tss.ss0 != 0)

                { /\* 左Ctrl+c \*/

                    cons = (struct CONSOLE \*)\*((int \*)0x0fec);

                    cons\_putstr0(cons, "\nBreak(key) :\n");

                    io\_cli(); /\*不能在改变寄存器值时切换到其他任务\*/

                    task\_cons->tss.eax = (int)&(task\_cons->tss.esp0);

                    task\_cons->tss.eip = (int)asm\_end\_app;

                    io\_sti();

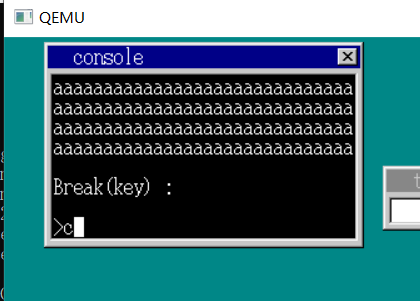
                }

将汇编中end\_app改为asm\_end\_app,用到的地方都需要修改，.h中声明

修改

struct TASK \*task\_alloc(void)

测试，按 左Ctrl+c即可结束程序



编写一个C语言输出字符串hello4和汇编输出hello5

显示窗口的winhelo和winhelo2

在a\_nask.nas中编写相应API

修改

int cmd\_app(struct CONSOLE \*cons, int \*fat, char \*cmdline);

int \*hrb\_api(int edi, int esi, int ebp, int esp, int ebx, int edx, int ecx, int eax);

主函数中加入

    \*((int \*)0x0fe4) = (int)shtctl;

修改Makefile

测试

