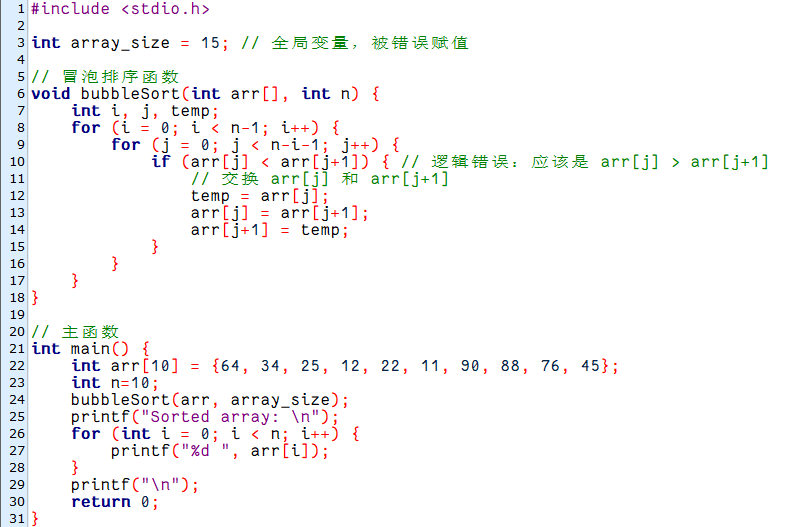
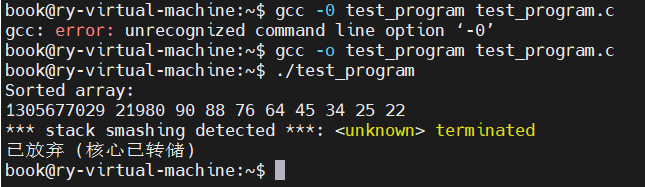
# 实验设计：ELF文件与虚拟内存

## 修改ELF文件(在ubuntu中修改，liteos环境下运行)

1. 这部分实验研究如何针对ELF可执行文件实现代码注入，这是一个简单的实例，但却可以帮助理解ELF文件的格式。

首先我们先编写一段代码，然后将其编译成可执行的ELF文件：

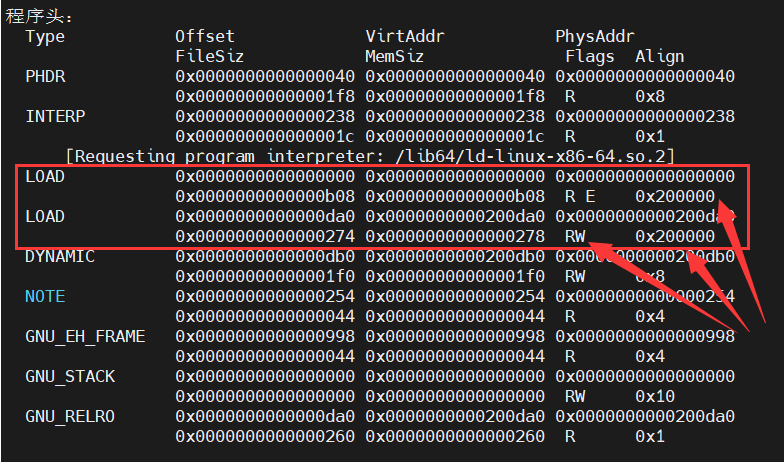
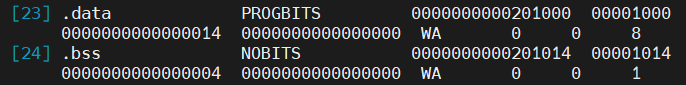




1. 当运行这个ELF文件时，我们发现程序时有越界问题，那是因为array\_size的值被赋为15。array\_size是一个全局变量，因此它应该存储在data段，我们对它进行修改。

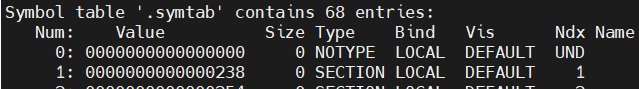
第一步：首先查看elf头部信息：

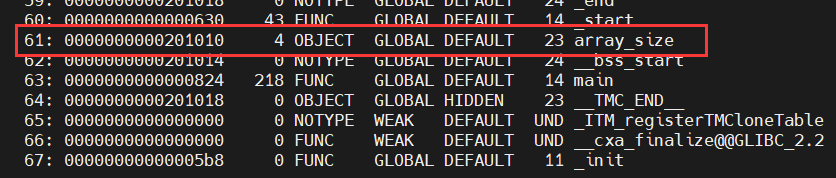




可以发现，data段被映射到0x201000处，而相对于ELF文件头部的偏移量是0x1000

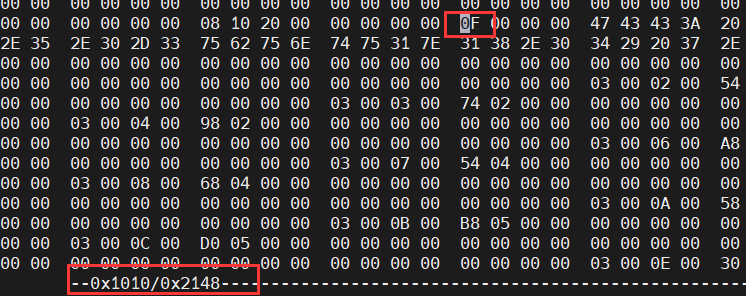
接着查看符号表，找到array\_size:



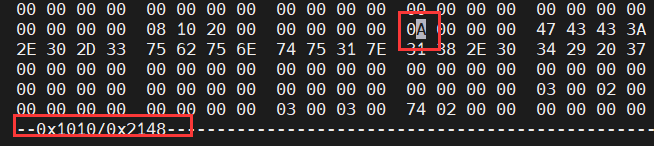


由此，我们可以计算出array\_size在ELF文件中的位置是0x1010。

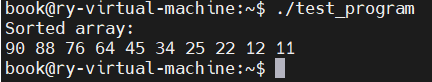
第二步：使用Hexedit打开ELF文件，找到其中array\_size的位置：



将其修改为10，对应16进制0x0A

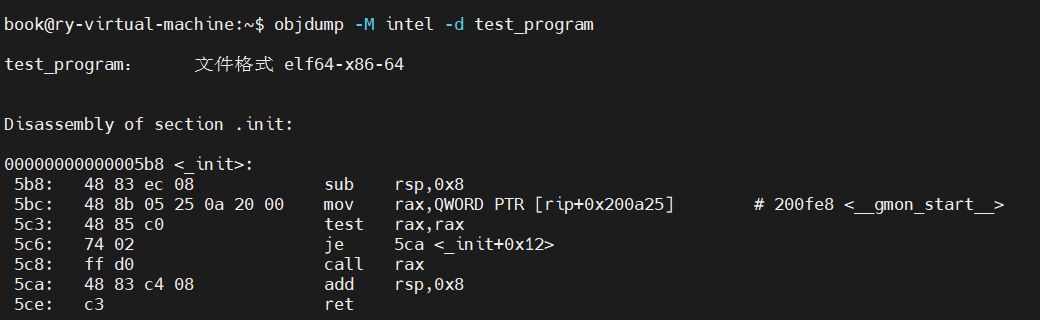


第三步：再次运行ELF文件，正确运行：

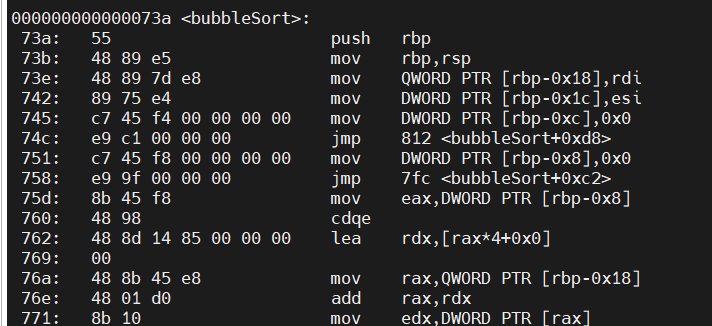


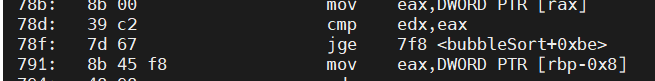
1. 这个程序目前实现的是降序排序，但如果想改为升序排序，应该怎么做呢？那就应该对ELF文件中的text段(代码段)进行修改。

第一步：对ELF文件进行反汇编：



查看BubbleSort函数的汇编代码：

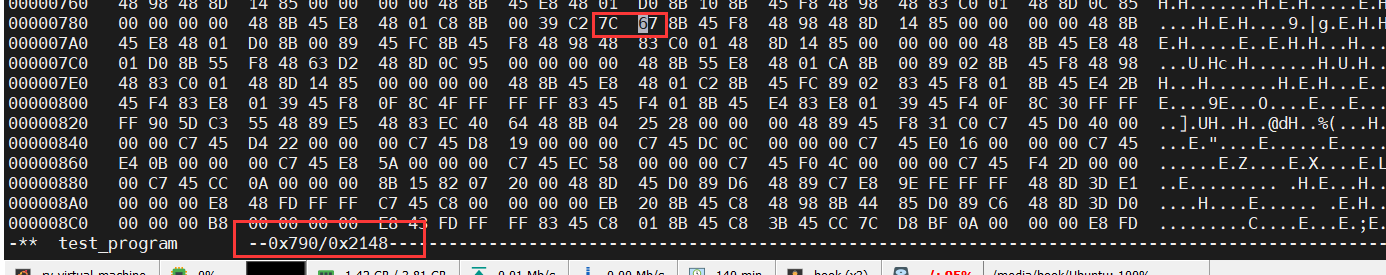




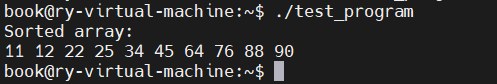
在这里，cmp edx, eax 指令比较 edx 和 eax 寄存器的值，这两个寄存器分别加载了 arr[j] 和 arr[j+1] 的值。紧接着的 jge 指令（Jump if Greater or Equal）是一个条件跳转，它基于比较的结果来决定是否跳转到指定的地址。在当前的降序排序实现中，如果 arr[j] 大于等于 arr[j+1]（即 edx >= eax），则跳转继续下一个循环迭代。

要改为升序排序，我们需要在 arr[j] 小于 arr[j+1] 时进行交换。因此，您需要更改 jge 指令为 jl 指令（Jump if Less），它会在 arr[j] 小于 arr[j+1] 时跳转。机器码中，jge 指令的操作码是 7d，而 jl 指令的操作码是 7c。

第二步：使用Hexedit打开ELF文件，进行修改：



第三步：再次运行ELF文件：



## 观察ELF如何映射到虚拟内存