实验5

班级：191211

姓名：孙鹤轩

学号：20211000156

实验内容：

1 缓冲区溢出攻击实验的内容、原理、方法和基本步骤；

2 过程调用的机器级表示、栈帧组成结构、缓冲区溢出等知识的回顾与应用。

实验目标：

1 加深对函数调用规则、栈结构、缓冲区溢出攻击原理、方法与防范等方面知识的理解和掌

握；

2 从程序员角度认识计算机系统，将程序设计、汇编语言、系统结构、操作系统、编译链接

中的重要概念贯穿起来，对指令在硬件上的执行过程和指令的底层硬件执行机制有深入的理

解；能够以需求分析为基础，对计算机系统模块或单元进行操作。

3 掌握各种开源的编译调试工具。

实验任务：

1 学习 MOOC 内容

https://www.icourse163.org/learn/NJU-1449521162

第六周 缓冲区溢出攻击

第 1 讲 缓冲区溢出攻击实验：概述

第 2 讲 缓冲区溢出攻击实验：目标程序与辅助工具

第 3 讲 缓冲区溢出攻击实验：Level 0

第 4 讲 缓冲区溢出攻击实验：Level 1 及课后实验

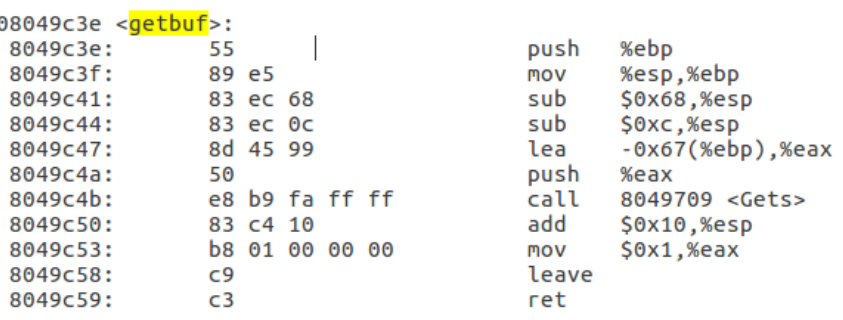
2 完成实验

详见缓冲区溢出攻击实验文档

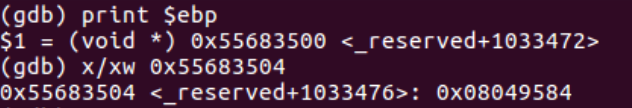
2.1 第一关 smoke

2.1.1实验结果及分析

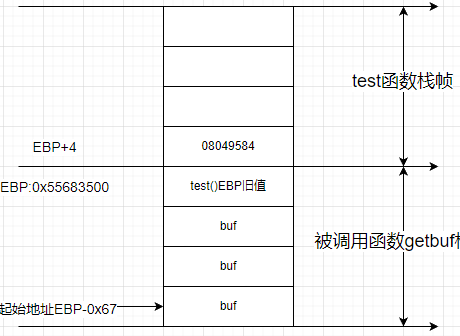
首先我们获得getbuf函数的汇编代码，如下：



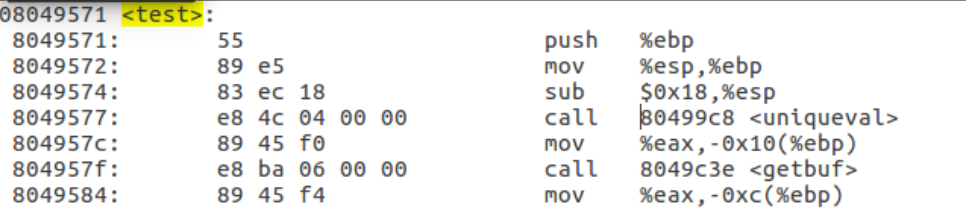
接着确定buf数组的首地址，根据8049c47-8049c4b处指令可以看出eax的值作为Gets函数参数输入，Gets函数的输入参数即为buf数组的首地址，即-0x67(%ebp)处，我们需要将test函数的返回地址设置为smoke函数的起始地址，设置断点在call指令前来观察确定栈帧结构：

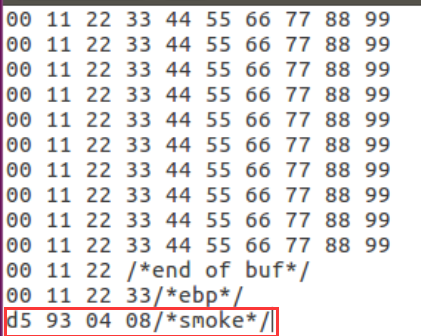


可以看到ebp的地址是0x55683500，栈帧结构如下：

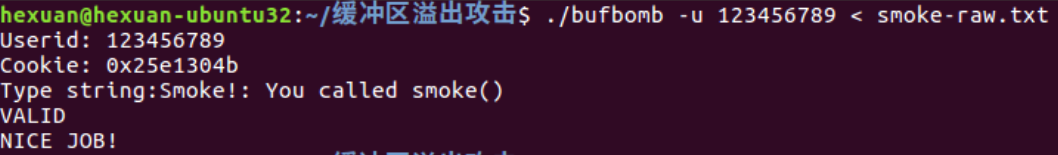


那么test函数的返回地址应为0x55683500+4=0x55683504地址处的值，gdb查看为0x08049584，根据程序调用过程，该返回地址为call指令的下一条指令地址，我们来看test函数的反汇编代码：

其地址确实是8049584，所以我们需要将这个地址改为smoke的地址，我们对buf数组进行填充，buf首地址与返回地址相差 0x67+4即107个字节，所以我们填充如下字符串：



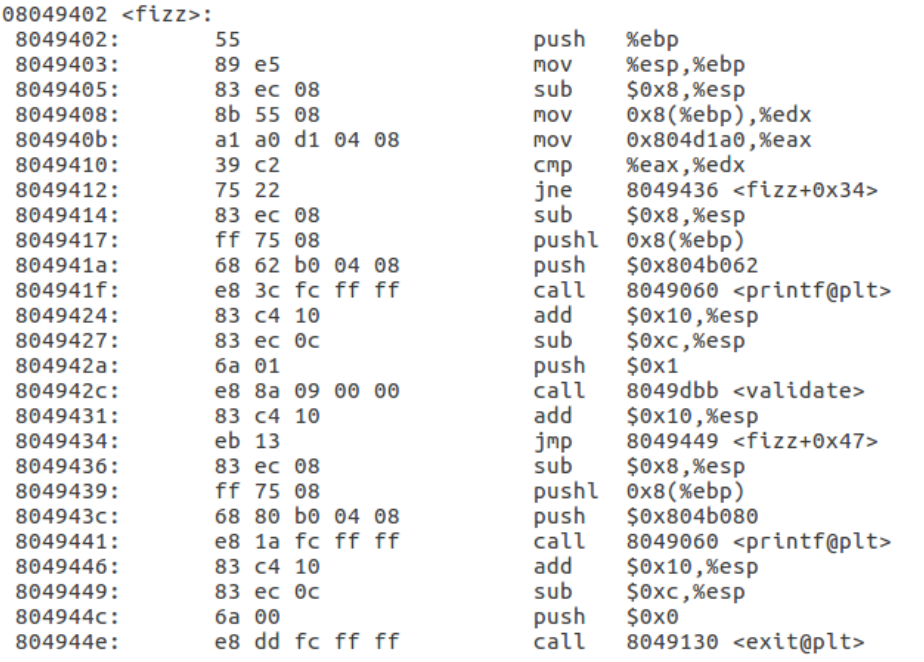
将最后四个字节设置为smoke函数的首地址，试验结果如下：



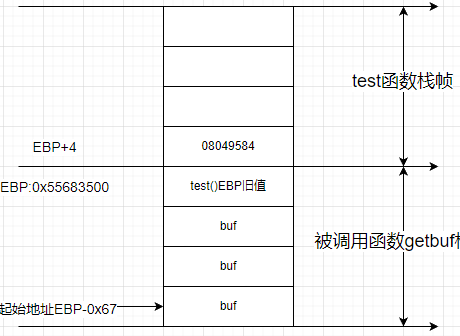
2.2 第二关 fizz

2.2.1实验结果及分析

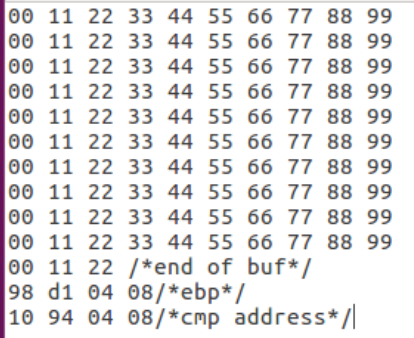
首先我们获得fizz函数的反汇编代码，如下：



我们需要使cmp的比较结果为真来执行我们的目标代码，即让eax的值等于edx的值，eax的值为0x804d1a0地址处的值应为全局静态变量常熟cookie的存储地址，0x8（%ebp）为我们输入的值，所以我们的思路是将0x8(%ebp)处的地址改为全局变量cookie的地址，让其值相等，栈帧结构分析如下：



与上一题不同，我们需要更改EBP的值让其在调用fizz函数时的地址为cookie地址-0x8即0x804d1a0-0x8=804d198，同时将返回地址设置为cmp指令的地址即：8049410即可跳转到我们想要执行的的代码段，所以按照如下思路构建字符串：



实验结果如下：

