

深入理解计算机系统》lab3)

阅读数：10603 标签：32位 计算机 csapp buflab 更多



漏洞干一些非法的事情（狐狸脸ing）的lab。。。

lab简单一些，只要弄清楚buffer的原理还是比较容易做的。

\*\*\*\*\*

函数，程序利用这个函数来建立buffer

```

    push    %ebp
    mov     %esp,%ebp
38         sub     $0x38,%esp
d8         lea     -0x28(%ebp),%eax<-buffer共0x28byte
24         mov     %eax,(%esp)
ff ff ff  call     8048bf1 <Gets>
00 00 00  mov     $0x1,%eax
         leave
         ret
```

ved %esp)构成了getbuf的栈结构，具体结构图如下：



\*\*\*\*\*

to execute the code for smoke when getbuf executes its return statement,rather than returning to test

push %ebp

mov %esp,%ebp

👑 VIP

免广告

^

🛡

\*\*\*\*\*

o execute the code for fizz rather than returning to test.

```

    push    %ebp
    mov     %esp,%ebp
18      sub     $0x18,%esp
08      mov     0x8(%ebp),%eax
c4 b1 04 08    cmp     0x804b1c4,%eax
           jne     8049158 <fizz+0x2f>
24 04      mov     %eax,0x4(%esp)
```

<fizz>函数，还需要在0x8(%ebp)中放0x804b1c4中存的数值。查找到这个地址中的数值，发现就是cookie值（po主cookie是0x51ade980），那么只需要把0x51ade980放入0x8(%ebp)中即可。

0x51ade980在retaddr，那么0x8(%ebp)就是在栈顶下面的位置。

\*\*\*\*\*

o execute the code for bang rather than returning to test

```

    push    %ebp
    mov     %esp,%ebp
```

```
18      sub    $0x18,%esp
c4 b1 04 08      cmp    0x804b1c4,%eax
jne      804910d <bang+0x31>
```

g>函数，要把内存中的0x804b1cc 中的值取出来与0x804b1c4(这个上一题已经得出是cookie值) 比较，查看0x804b1cc，发现是一个<.global\_value>,那么就写一段攻击代码如下:

-更改global\_value的值  
bang作为retaddr压栈

二进制代码，填入buffer，注意，还要查找到the start of input string来作为第一次retaddr，那么程序就从第一次return到我们输入的字符串，然后执行我们的攻击代码，再进入<bang>,完成任务。  
uf>中设置断点（关于gdb使用请参考上一篇博文，关于bomb lab的那篇），查找-0x28（%ebp)的地址（这个地址是我们输入的buffer区的起始位置）

\*\*\*\*\*

loit string that will cause getbuf to return your cookie back to test, rather than the value 1.

```
      push    %ebp
      mov     %esp,%ebp
      push    %ebx
24      sub     $0x24,%esp
fe ff ff      call    8048bd0 <uniqueval>
f4            mov     %eax,-0xc(%ebp)
ff ff ff      call    8048ca4 <getbuf>
      mov     %eax,%ebx
```

e而不是原来的0x1.那么同上题一样的思想，我们写一段攻击代码

外，这个题因为是要按正常方式返回原函数，那么我们要保证saved ebp的值是正确的。saved ebp就是<getbuf>中的ebp值，同样使用gdb设置断点调试可以得到。按照栈结构将所得到的数据输入即可得到答案。

\*\*\*\*\*

Dynamite level. Once again, your job for this level is to supply an exploit string that will cause getbufn to return your cookie back to test, rather than the value 1.

<getbufn>函数，会使得栈底在一定范围内变化5次。

```

                                push    %ebp
                                mov     %esp,%ebp
18 02 00 00                    sub     $0x218,%esp
f8 fd ff ff                    lea     -0x208(%ebp),%eax
24                             mov     %eax,(%esp)
ff ff ff                      call    8048bf1 <Gets>
00 00 00                      mov     $0x1,%eax
                                leave   %eax
                                ret
```

x4的，结构同上。整个buffer共有528byte。

```

                                push    %ebp
                                mov     %esp,%ebp
                                push    %ebx
24                             sub     $0x24,%esp
f4 ef be ad de                movl    $0xdeadbeef,-0xc(%ebp)
ff ff ff                      call    8048c86 <getbufn>
                                mov     %eax,%ebx
```

先要复原%ebp内容，从<testn>函数中看出来这时的ebp应当是esp+0x24+0x4(push ebx)=esp+0x28.然后在将cookie赋值填入eax中

代码此外，由于每次栈的不确定性，我们需要先空执行程序，利用上面同样的方法查找到每次的start of input string。找到5个之后，（可能有相同的，不影响结果，这个可以认为是有两次执行的栈结构相同），为选择最大的一个数据作为我们整个攻击代码的start of input string。至此，我们解决了大部分问题，最后，为了使每一个开始的位置都能进入我们的attack code 我们需要把文件的其他地方用nop(90)填充，这样才能攻击代码中。得到答案：





e 得到

赚钱新方法！轻松月入高薪！

收几股，千万别卖，或将大涨！

常感谢你把解题思路分享出来。有个小疑问，讨论一下。level 1中最后填充的8个字节00 00 00 00 80 e9 ad 51，是因为传入到fizz函数的参数是int，占用8个字节。然后，在8个字节内0在前），而不是小端模式（00 00 00 00在后）排列。 （2年前 #3楼） [查看回复\(1\)](#)

level4里面的攻击代码是 leal -0x28(%esp),%ebp ? 还是leal 0x28(%esp),%ebp (正数还是负数) （3年前 #2楼） [查看回复\(1\)](#)

法我还是有点奇怪.buffer是在getbuf栈内的，而当leavlq 和ret之后getbuf的栈就会被销毁，然而return address已经被改成栈被的地址，这样一来，岂不是逻辑矛盾？ （4年前 #1楼）

之路 4533 来自： [未选之路](#)

out(缓冲区溢出实验) - The\_V\_的博客 1.2万 来自： [The\\_V\\_的博客](#)

Jason Leaster | Rebuilding the tower of b... 4566 来自： [Jason Leaster | ...](#)

什么区别 广告

区溢出攻击 《深入理解计算机系统》 - fang92... 2703 来自： [fang92的专栏](#)

新颜\_USTC 589 来自： [新颜\\_USTC](#)

bufbomb - qq\_15514565的博客 820 来自： [qq\\_15514565的...](#)

（二） - 等价交换 1244 来自： [等价交换](#)

Hardware/Software Interface》的实验三。Leve...





《深入理解计算机系统》笔记-第三章 程序的机器级表现 - HHL... <div>2.c gcc: GCC C编译器 (or cc启动) -01: 使...</div>		👁 1154	来自: <a href="#">HHLab</a>
- 漫长的旅途 <div>/code.html</div>		👁 1304	来自: <a href="#">漫长的旅途</a>
- 一名渣程序员的无厘头记录		👁 980	
系统》 (Computer System: A Programmer's ...		来自: <a href="#">一名渣程序员的...</a>	
飞龙			07-04
弹《深入理解计算机系统》 - fang92的专栏		👁 4409	
进行反汇编，得到一个1000+的汇编程序，看...		来自: <a href="#">fang92的专栏</a>	
年追涨停铁律“1272”曝光，震惊众人			
后一物，体重不过百！呼和浩特人必看！			
N博客			
要。 Buffer=数据+四个索引 正是四个索引才是的数据可以高效访问,这四个索引是:mark(标记),position(位置),limit(界限),...			
入理解计算机系统》lab7) -..._CSDN博客			
机系统》lab2) 阅读量:12100 <csapp> buffer lab (《深入理解计算机系统》lab3) 阅读量:10148 <csapp> data lab (《深...			
N博客			
要。 Buffer=数据+四个索引 正是四个索引才是的数据可以高效访问,这四个...			
入理解计算机系统》lab7) -..._CSDN博客			
机系统》lab2) 阅读量:12100 <csapp> buffer lab (《深入理解计算机系统》l...			
aojiang的博客		👁 3615	
组 Cache main函数 选项参数处理 读指令模拟...		来自: <a href="#">mo_xiaojiang的...</a>	
			06-03
深入理解计算机系统》lab7) - Stone		👁 3037	
。一直没有更新。。 好了现在开始慢慢更一下...		来自: <a href="#">Stone</a>	
理解计算机系统》lab1) - Stone		👁 4649	
		来自: <a href="#">Stone</a>	
out (深入了解计算机系统 实验一) - lzjsqn的...		👁 6318	
Data Lab * * * * bits.c - Source file with your sol...		来自: <a href="#">lzjsqn的专栏</a>	
赚钱新方法！轻松月入高薪！			
赚钱新方法！轻松月入高薪！			
解计算机系统 实验二) - lzjsqn的专栏		👁 4616	
做笔记吧，实在写不动了！1.执行反汇编 obj-du...		来自: <a href="#">lzjsqn的专栏</a>	
入理解计算机系统》lab6) (附lab4\lab5下载...		👁 3536	
		来自: <a href="#">Stone</a>	

机系统) 第三版 实验 - qq_29719481	04-21
料! 经典教材深入理解计算机系统最新版实验材料!	
人理解计算机系统》lab2) - Stone	👁 1.2万
dn.net/download/u013648407/7279933 其中bo...	来自: Stone
程序性能优化实验 - syq1207的博客	👁 6453
, 昨天周六下午刚刚验收完所带班级的必做实验...	来自: syq1207的博客
赚钱新方法! 轻松月入高薪!	
了, 2018聪明的呼和浩特人都在靠它赚外快	
之路 - CSDN博客	
机系统》lab3) - Stone 05-14 1万 lab3 buflab。一个训练你利用buffer漏洞干一些非法的事情(狐狸脸ing)的 来自: Stone...	
博客 - CSDN博客	
长度的Buffer实例 let a: Buffer = new Buffer(number: length); # 为Buffer实例赋值 a.fill(value); 通过数组实现Buffer的...	
之路 - CSDN博客	
机系统》lab3) - Stone 05-14 1万 lab3 buflab。一个训练你利用buffer漏洞干...	
博客 - CSDN博客	
长度的Buffer实例 let a: Buffer = new Buffer(number: length); # 为Buffer实...	
原创(北大&cmu;) 全集ABC 仅供参考, 请勿抄袭 - braveryCHR	12-27
全集ABC 仅供参考, 请勿抄袭	
	👁 418
bj/article/details/70156819 实验二链接 http://b...	来自: scaubj的博客
; Buflab实验, 缓冲区溢出攻击实验(1) - sy...	👁 6579
感觉之前做过那个bomb实验以后, 这个实验相...	来自: syq1207的博客
聚 - Yiyang的专栏	👁 1.1万
栈性质的缓冲区溢出实验, 能够帮助你加深理解...	来自: Yiyang的专栏
详细过程(内含源程序包及文档)	04-23
耐基 梅隆大学)经典计算机课程实验之一, 里面含有实验完整内容及其源程序, 还有详细的解答过程, 很多国内大学的计算机课程都选用此...	
工挣两三千? 她们用微信就能挣翻倍的!	
持25岁美女用手机做这个, 1个月存款吓呆父母!!	
记录 - 俯瞰风景	👁 195
l 0: 要求getbuf()执行完后, 跳到smoke()里面 0...	来自: 俯瞰风景
uffer Bomb)解答及实验报告	04-13
我的解答及实验报告	
第三版 csapp 3ed	01-13
ab - peanWang的博客	👁 196
三: 工具 四: 实验内容	来自: peanWang的博客

系统 proxy lab - donggua\_fu的博客 472

第一步：深度参考tiny.c，再自己改改就行，验... 来自： donggua\_fu的...

外卖小哥辞去工作三个月，惊人存款曝光！

妈妈离婚后只因这个，过得更好了，还送孩子上了国际学校！

【Buflab实验,缓冲区溢出攻击...\_CSDN博客

dout.tar.gz文件夹拖到我们的虚拟机的csapp...<csapp> buffer lab (《深入理解计算机系统》lab3) 05-14 1万 lab3 buf...

【Buflab实验,缓冲区溢出攻击...\_CSDN博客

dout.tar.gz文件夹拖到我们的虚拟机的csapp...<csapp> buffer lab (《深入理...

5) - 小白的博客 1562

来自： 小白的博客

PP》练习题笔记（一） - 鸟恋旧林的博客 3527

习题笔记（一） 来自： 鸟恋旧林的博客

CSAPP) 英文原版(完整版!!!) 05-13

这个是完整的,而且 第四章编排到书签目录里面了.

【Buflab实验，缓冲区溢出攻击实验(2) - syq1... 4067

以及smoke函数的c代码如下： Level0 就很简... 来自： syq1207的博客

ne-R 05-13

去 以及所有需要的软件 a\*.txt是po主答案 在po主blog中有详解

什么区别



【Buflab实验，缓冲区溢出攻击实验(4) - syq1... 4162

返回到test，而是去执行函数bang，但是区别是... 来自： syq1207的博客

azard5的专栏 1153

课程中CSAPP那本书上的6个lab。 Lab 1 :... 来自： azard5的专栏

ZZY的博客 1950

来自： PKU\_ZZY的博客

Jason Leaster | Rebuilding the tower of b... 5656

来自： Jason Leaster | ...

未选之路 4459

来自： 未选之路



pp

引发的思考 - Jason Leaster | Rebuilding th... 2538

来看邮件发现有同学和我讨论关于函数调用压栈... 来自： Jason Leaster | ...



**S-tone-R**

关注

原创

5

粉丝

15

喜欢

1

评论

8

等级：

博客 2

访问：

3万+

积分：

370

排名：

23万+

想开发一个app

摄像机推荐

广告

最新文章

<csapp> pipeline lab （《深入理解计算机系统》lab7）

<csapp> malloc lab （《深入理解计算机系统》lab6）（附lab4\lab5下载地址）

<csapp> bomb lab （《深入理解计算机系统》lab2）

<csapp> data lab （《深入理解计算机系统》lab1）

个人分类

ics-lab

6篇

归档

2015年1月

1篇

2014年8月

1篇

2014年5月

2篇

2014年4月

2篇

热门文章

<csapp> bomb lab （《深入理解计算机系统》lab2）

阅读量：12385

<csapp> buffer lab （《深入理解计算机系统》lab3）

阅读量：10601

<csapp> data lab （《深入理解计算机系统》lab1）

阅读量：4649

<csapp> malloc lab （《深入理解计算机系统》lab6）（附lab4\lab5下载地址）

阅读量：3535

最新评论

<csapp> bom...

qq\_36570544： 能详细说说第二个怎么拆吗

<csapp> bom...

qq\_36570544： 这个phase\_2和1一样呀

<csapp> buf...

u013648407： [reply]guokaiwhu[/reply] int是4位 little endian只针对地...

<csapp> buf...

guokaiwhu： LZ，你好。非常谢谢你把解题思路分享出来。有个小疑问，讨论一下。 level 1中最后填充的8个字...

<csapp> buf...

u013648407： [reply]PleaseStandByMe[/reply] 负数。 栈结构是与一般理解反的，es...



联系我们



微信客服



QQ客服

 QQ客服


 kefu@csdn.net

 客服论坛

 400-660-0108

工作时间 8:00-22:00

关于我们 | 招聘 | 广告服务 | 网站地图

 百度提供站内搜索 京ICP证09002463号

©1999-2018 江苏乐知网络技术有限公司

江苏知之为计算机有限公司 北京创新乐知信息技术有限公司版权所有


网络110报警服务    经营性网站备案信息


北京互联网违法和不良信息举报中心


中国互联网举报中心


联系我们



 QQ客服

 kefu@csdn.net

 客服论坛

 400-660-0108

工作时间 8:00-22:00

关于我们

|


招聘

|

广告服务

|

网站地图

 百度提供站内搜索 京ICP证09002463号

©1999-2018 江苏乐知网络技术有限公司

江苏知之为计算机有限公司 北京创新乐知

信息技术有限公司版权所有

网络110报警服务

经营性网站备案信息

北京互联网违法和不良信息举报中心

中国互联网举报中心