stack-overflow

task1:

手机屏幕的截图

描述已自动生成

task2:

手机屏幕截图

描述已自动生成

电脑屏幕的照片上有文字

描述已自动生成屏幕上有字

描述已自动生成

从上图可以看到，漏洞程序读取badfile 文件到缓冲区str，且str的地址为0xbffff437，计算上shellcode偏移量100（0x64）,则shellcode地址为0xbffff49b。偏移量为0x24

所以exploit.c代码中加入

﻿strcpy(buffer+100,shellcode);

strcpy(buffer+0x24,"\x9b\xf4\xff\xbf");

task3:

电脑屏幕截图

描述已自动生成电脑屏幕的照片

描述已自动生成

可以看到当去掉setuid(0)前的注释后，获得的shell的权限变成root用户了。

电脑屏幕的照片上有文字

描述已自动生成

可以看到此时报一个错误是因分配空间不足引起的"stack smashing detected"。

Return-to-libc Attack Lab

电脑屏幕截图

描述已自动生成

关闭地址随机化，编译漏洞文件，注意设置栈不可执行但要关闭栈保护，并且赋予其SUID权限。

电脑屏幕的照片上有文字

描述已自动生成屏幕上有字

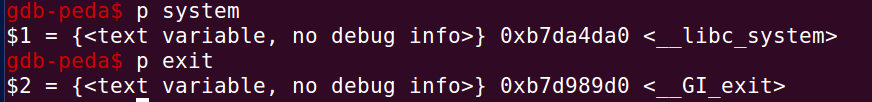
描述已自动生成手机屏幕截图

描述已自动生成图片包含 游戏机, 画

描述已自动生成



使用gdb调试查看返回地址离fread写入处的距离。bof正常返回地址0x08048555，badfile(CCCD)和bof函数正常返回地址相差24个字节。



使用gdb查看函数的地址

图片包含 标志, 游戏机, 食物

描述已自动生成

使用geeenvaddr.c获取环境变量的地址。

所以exploit.py中X=24,Y=28,Z=32

手机屏幕截图

描述已自动生成

手机屏幕截图

描述已自动生成

实验成功