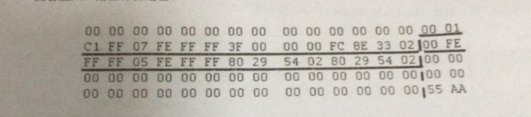
2015-软件安全

一、计算题(每题5分，共20分)

1、以下是某硬盘的分区表信息，计算所有分区大小。(按1000计算， 小数点后取1位，四舍五入，给出计算过程)



0x02338efc=36933372

36933372\*512/10^9=18.9G

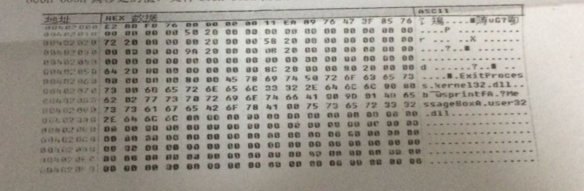
0x02542980=39070080

39070080\*512/10^9=20.0G

2. 下图为某程序的.rdata节(开始位置RVA: 2000， 文件偏移量: 800H)在内存中的主委

数据。试分析计算MessageBox 函数的真实地址，wsprintfA 函数的真实地址，该文件

800H-803H偏移处的值，文件808H- 80BH偏移处的值。(给出计算思路和结果)



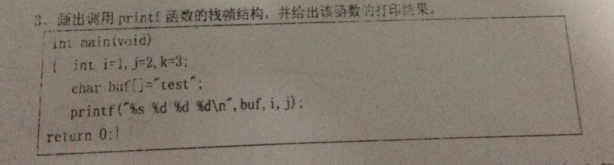
Messagebox:0x7689ea11

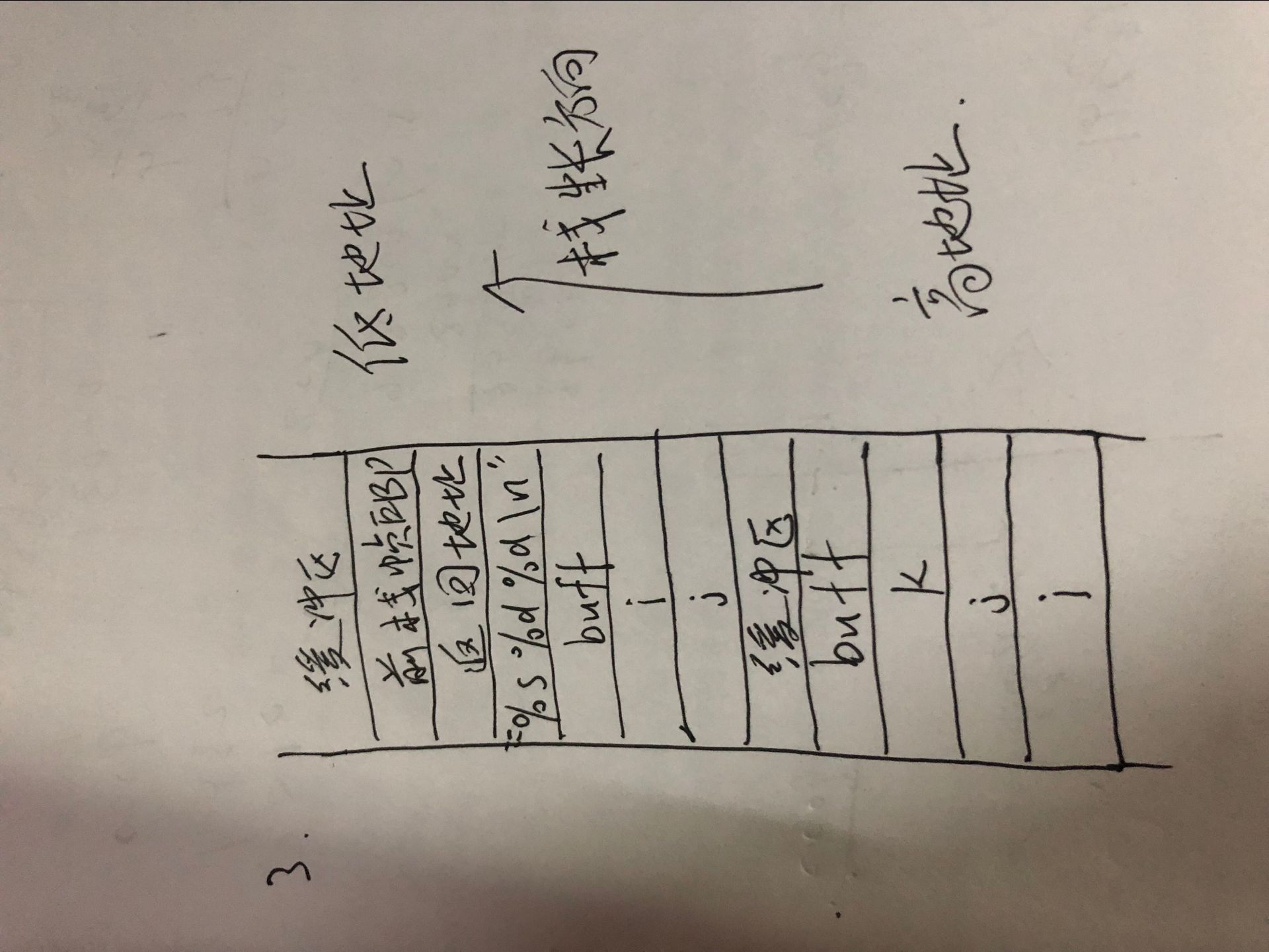
wsprintfA:0x76853f47

800H-803H:0x00002064

808H- 80BH: 0x0000208c

3.通出调用printf函数的栈帧结构，并给出该函数内打印结果。





test,1,2,1953719668

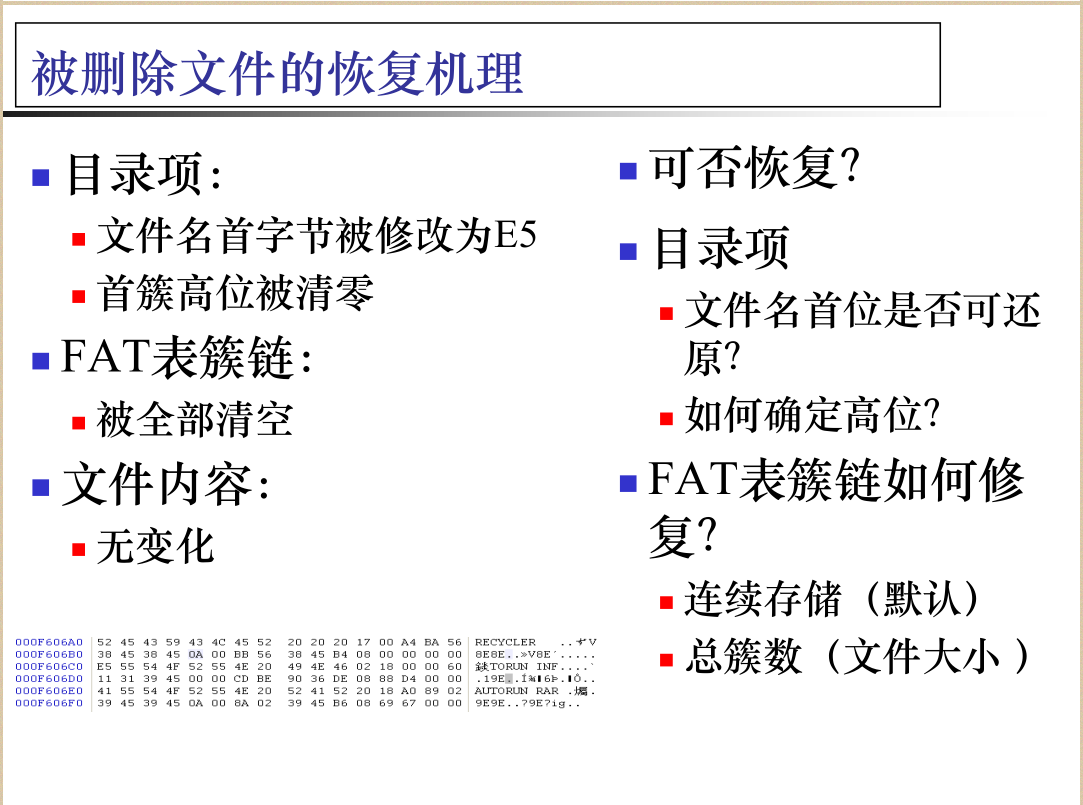
4 FAT32中，一个8kb目录空间通常可以包含多少个目录项?当该目录空间占满时，如何扩充该目录空间?

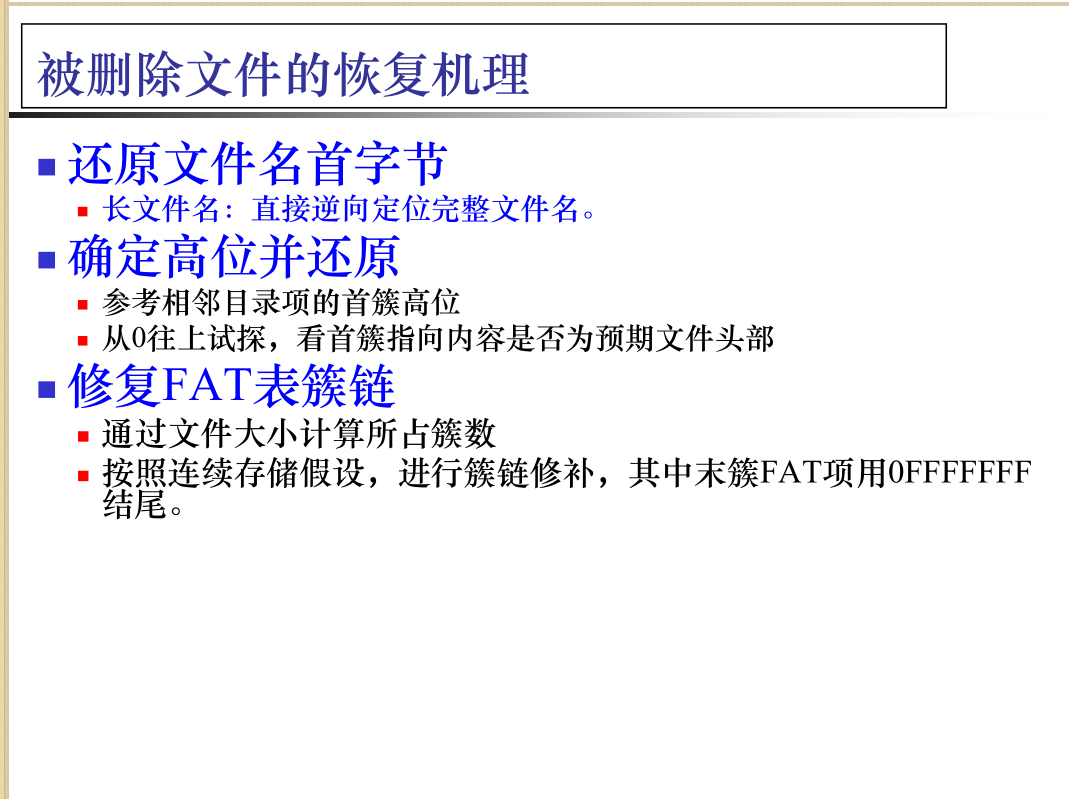
8kb/32b=256个目录项

文件系统会另行分配簇号用于存储目录项，根目录所对应FAT表项会写入又分配的簇的簇号

二．简答题

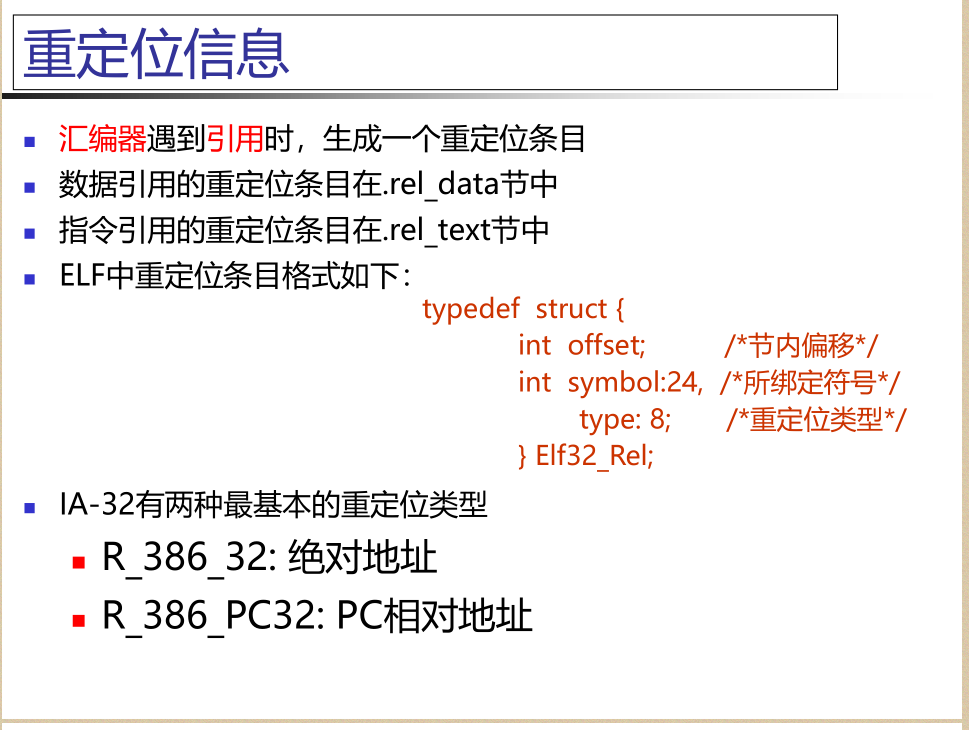
1. U盘比较常见的文件系统格式是FAT32， 假设其中存储的一个DOC文件被无意删除，试给出手动恢复的原理和过程。





1. 请描述PE结构中重定位节的结构，以及重定位的作用。





3.“攻击者使用网络木马进行远程文件操作”与“用户自身使用资源管理器进行本地文件操作”存在多方面的差别，请给出可用于在用户计算机中检测区分这两类活动的技术特征。

木马的文件管理 P212

文件资源管理器是一项系统服务，负责管理数据库、持续消息队列或事务性文件系统中的持久性或持续性数据。资源管理器存储数据并执行故障恢复. “文件资源管理器”是Windows 系统提供的资源管理工具，我们可以用它查看本台电脑的所有资源，特别是它提供的树形的文件系统结构，使我们能更清楚、更直观地认识电脑的文件和文件夹。另外，在“资源管理器”中还可以对文件进行各种操作，如：打开、复制、移动等。

1. 计算机病毒的文件感染与漏洞攻击的Shellcode注入有哪些异同?

计算机病毒（Computer Virus）是编制者在计算机程序中插入的破坏计算机功能或者数据的代码，能影响计算机使用，能自我复制的一组计算机指令或者程序代码。

shellcode实际是一段代码（也可以是填充数据），是用来发送到服务器利用特定漏洞的代码，一般可以获取权限。另外，Shellcode一般是作为数据发送给受攻击服务器的。

5.请描述Windows下至少3种不同的软件行为拦截方法和实现层次，并简要说明其优缺点。

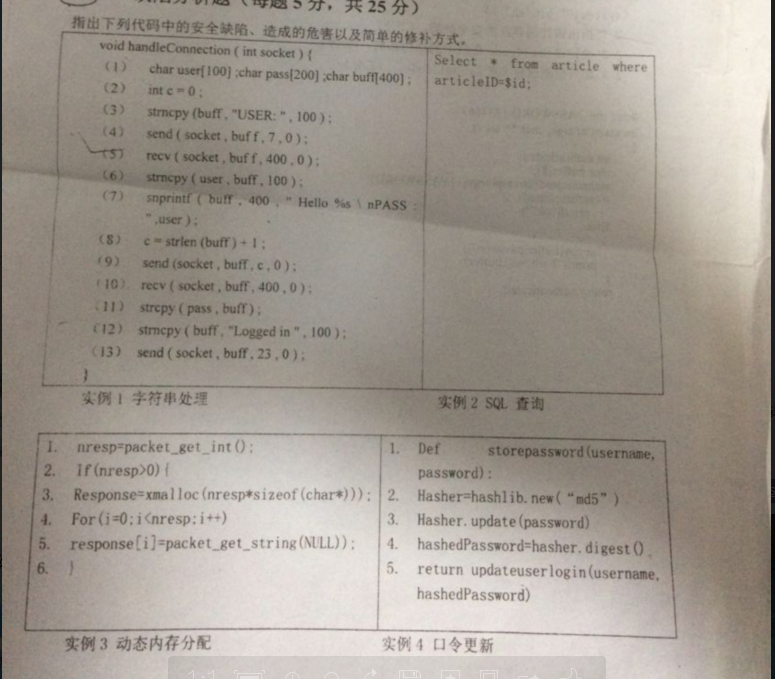
没百度到 教材P275

6.如何对一款远程控制型网络木马的控制者进行溯源追踪?

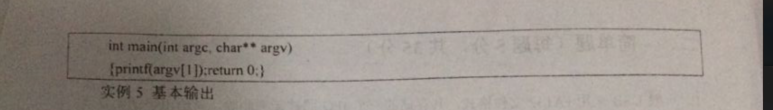
7.ASLR、DEP、GS、SafeSEH是常见的四种漏洞利用阻止技术，分别用于于扰或阻止漏洞利用流程中的特定环节，如果按照其对应的特定环节在漏润利用流程中发生作用的先后顺序来排列，其顺序应该是怎样的?并请说明理由。

三．缺陷分析题

指出下列代码中的安全缺陷、造成的危害以及简单的修补方式。



3) 上面这个代码中，nresp是size\_t类型，这个示例是一个解数据包的示例，一般来说，数据包中都会有一个len，然后后面是data。如果我们精心准备一个len（对应代码中的nresp），比如：1073741825（在32位系统上，指针占4个字节，unsigned int的最大值是0xffffffff，我们只要提供0xffffffff/4 的值——0x40000000，这里我们设置了0x4000000 + 1）， nresp就会读到这个值，然后nresp\*sizeof(char\*)就成了 1073741825 \* 4，于是溢出，结果成为了 0x100000004，然后求模，得到4。于是，malloc(4)，于是后面的for循环1073741825次，就可以干坏事了（经过0x40000001的循环,用户的数据早已覆盖了xmalloc原先分配的4字节的空间以及后面的数据，包括程序代码，函数指针，于是就可以改写程序逻辑。  
4)第四题用的是未加salt的单向哈希函数加密，攻击者可以通过查表法，逆向查表法去破解密码，导致密码的泄漏。  
解决方法：加上salt进行加密：hasher = hashlib.new('md5',b'SaltGoesHere')。



漏洞类型：格式化字符串漏洞

漏洞成因：printf函数本身无法确定参数的个数，而是通过格式化字符串的个数来依次访问栈中的变量进行匹配，如果在输出的字符串中嵌入格式化字符串，就会造成越界访问，并可以通过%n进行内存地址的重写。

利用方式：首先令argv[1]为一串%p(可以使用%$来选取特定参数)来打印出栈中的内容，然后在特定的位置加入%n(可以使用%$来选取特定参数)来重写相应地址的内容。

四、 综合设计题

1.栈溢出攻击经常会覆盖栈中的返回地址， /gs安全机制通过在函数入口往栈中压入一个随机度-cookie.函数出口检验该随机值，从而防御栈溢出。

1.1试设计一种对栈中返回地址备份的安全防御机制。

1.2与/gs相比，试分析该机制的优点与不足。

1.1在主函数中声明一个数组，在调用一个函数之后，把当前堆栈中的返回地址备份到数组中；在函数返回之前，用数组中备份的返回地址覆盖堆栈中的返回地址。

1.2优点：在gs防御机制中只能判断是否发生缓冲区溢出，而在真正发生溢出时不能进行保护。而该机制可以确保函数的返回地址是正确的，因此函数返回后可以正常执行剩余的流程不受到影响。

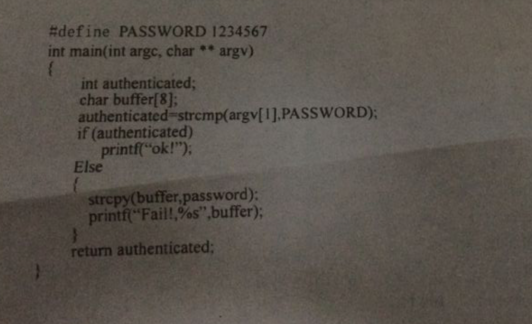
缺点：如果备份数组被溢出并修改为特定的地址，那么防御机制就会被攻破。

2、如果下列代码存在缺陷，试设计一个ROP的shellcode, 该Shellcode 弹出一个shell(systemy"/bin/sh" ))，

2.1.指出该代码存在的安全缺陷。

2.2设计该Shel lcode在栈中的结构布局。

2.3如何检测这种ROP的Shellcode?



a.安全缺陷：由于buffer的长度只有8，而password的长度没有限制，因此执行strcpy(buffer,password)函数时可能发生缓冲区溢出

b.payload:首先通过一串包含pop的gadget跳转到构造的ROP链，在ROP链中，首先在内存中搜索system函数的基地址，然后在bss段中写入"/bin/sh "字符串，最后将bss地址作为参数安排在相应的位置，调用system函数。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| junk | junk | junk | pop4\_ret | junk | junk | junk | pop4\_ret | 自己的ROP链 |

函数里的栈分布

main里的栈分布

c.检查关键函数的返回地址所指向的指令的前一条指令是否为call指令，如果不是则认为检测到ROP