170525 逆向-数学运算符的优化，CrackMe(2)

原创

奈沙夜影 于 2017-05-26 01:32:04 发布 425 收藏
分类专栏： CrackMe
版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。
本文链接：https://blog.csdn.net/whklhhhh/article/details/72760903
版权


CrackMe 专栏收录该内容
83 篇文章 2 订阅
订阅专栏

1625-5 王子昂 总结《2017年5月25日》【连续第236天总结】

A.加密与解密 数学运算符的优化

B.

加法：

一般情况下，加减法编译为add和sub指令。但是编译优化时，则经常使用lea指令

lea指令允许用户在一个时钟内完成c=a+b+78h计算，是非常快速的指令

eg:

```
push ecx
moveax,dword ptr [esp]
movecx,dword ptr [esp]
leaedx,dword ptr [ecx+eax+78] ;快速计算ecx+eax+78
```

lea命令等价于edx=ecx+eax+78

乘法：

乘法运算符一般编译成mul、imul指令。这些指令运行的速度比较慢，编译器会尽可能的提高代码的效率，因此倾向于使用别的

如果一个数是2的幂，那么会用左移指令shl来实现乘法

另外，加法对于提高3，5，6，7，9的乘法运算很有帮助

eg:eax*5可以写成 lea eax,[eax+4*eax]

ps:lea指令可以实现寄存器乘以2，4，8的运算

触发：

除法运算符一般编译成div、idiv指令。除法运算的速度就更慢了，大概比乘法多消耗10倍的CPU时钟

如果除数是2的幂，那么可以用较快的移位指令shr a,n来替换。移位指令只需要话费一个时钟，其中a是被除数，n是基数2的指数

除法指令需要用到符号扩展指令cdq，它的作用是把eax寄存器中的数视为有符号数，将其符号位扩充到edx寄存器。具体做法是

这样使得eax由32位带符号数变为了edx:eax的64位带符号数。

编译器优化时，会采用乘法字符串来代替除法运算。这样能提高效率。不过对于逆向来说会很难理解。

最常用的优化公式就是倒数相乘，即a/b=a*（1/b）

CrackMe（2）：
打开以后直接就是Name/Serial的序列号生成
运行，直接搜索字符串，在内存中找到判断跳转的指令
爆破，完成
注册机的查找参考了writeup：
一直向上翻，直到上一个retn后的push，一般就是一段代码的开头
下断点，然后单步运行一直跟，知道看到自己输入的Name出现后开始分析代码
本程序先是strlen取长度，然后乘以0x17CFB，再加上第一个字符的ANSI值，最后再经过一个vbaStrI4函数后，将得到的随机数与AKA-扑

vabStrI4经Writeup说明，是转十进制的函数
另外，大部分较难的程序直接查找字符串都是得不到结果的，因此应该跟API，这个才是无法避免的指令
本题刚开始我就用Ctrl+N查了，没有发现getMessageText或GetDialogText函数，猜测应该是VB编写的不需要？
Writeup提供了一个新的思路：在弹窗部分查堆栈
Alt+B进入堆栈窗口，查看函数调用，发现有两个MessageBox相关的函数，一个调用来自主线程，那没跑了，就跟它
跟过去就发现是错误弹窗的call，向上翻两行就看到了字符串的内容

B.明日计划

CrackMe(3)