

ctf.360.cn第二届，逆向部分writeup——第二题

转载

[weixin_33861800](#) 于 2014-11-26 14:05:51 发布 58 收藏

原文链接: <http://blog.51cto.com/cugou/1582761>

版权

题目: 见附件

这题目被提示的运行两次坑了。

说明: 已在压缩包中给定了一个用ReverseMe.exe加密过的文件-密文.db请分析ReverseMe.exe的算法, 写出解密算法, 解密该文件得到Key。该Exe里有一个bug, 导致exe无法运行;

提示:

你有两种方法得到该Key:

- 1.找到bug, patch掉之后, 运行两次该程序即可解密文件得到key。
- 2.老老实实的逆这个简单的算法, 写出一个解密程序, 解密。

OD载入, 设置命令行参数pass.db。

F8到VirtualAlloc函数, 发现size参数为0, 导致函数调用失败。

0040109E	. 6A 04	PUSH 4	Protect = PAGE_READWRITE AllocationType = MEM_COMMIT Size = 0 Address = NULL VirtualAlloc
0040109D	. 68 00100000	PUSH 1000	
004010A2	. FF35 FA2F4000	PUSH DWORD PTR DS:[402FFA]	
004010A3	. 6A 00	PUSH 0	
004010AA	. E8 E1000000	CALL <JMP.&kernel32.VirtualAlloc>	

根据pass.db长度, 设置size=0x10, F8继续。

发现CreateFileA的FileName参数依然是pass.db, 修改成pas1.db。

004010FE	. 6A 00	PUSH 0	hTemplateFile = NULL Attributes = NORMAL Mode = CREATE_ALWAYS pSecurity = NULL ShareMode = 0 Access = GENERIC_READ GENERIC_WRITE FileName = "pass.db" CreateFileA
00401100	. 68 80000000	PUSH 80	
00401105	. 6A 02	PUSH 2	
00401107	. 6A 00	PUSH 0	
00401109	. 6A 00	PUSH 0	
0040110B	. 68 000000C0	PUSH C0000000	
00401110	. 68 10324000	PUSH ReverseM.00403210	
00401115	. E8 64000000	CALL <JMP.&kernel32.CreateFileA>	

F9正常运行, 得到pas1.db文件。

实际上, 这里pas1.db的内容已经是答案了, 受到提示影响“patch掉bug后运行两次”, 所以得到pas2.db后, 怎么提交都不对, 后来发现pas2.db和pass.db的内容一样。才恍然大悟, 就是一个异或算法。

实际上这个所谓的加密算法就是和字符串“shit”进行异或运算。

转载于:<https://blog.51cto.com/cugou/1582761>