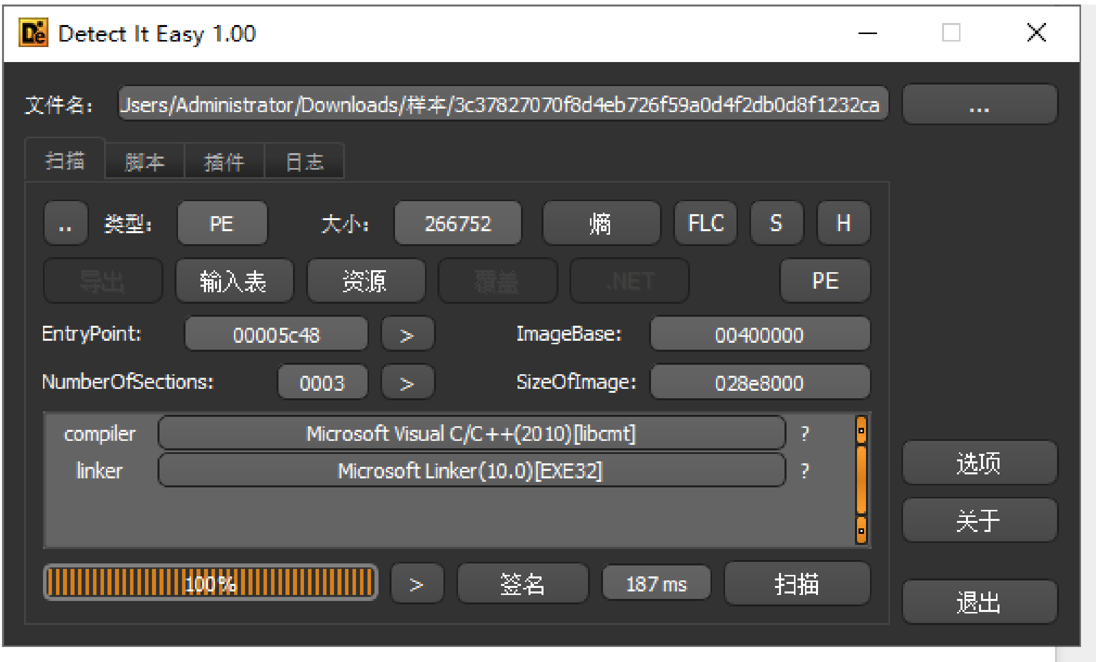
某恶意软件分析学习

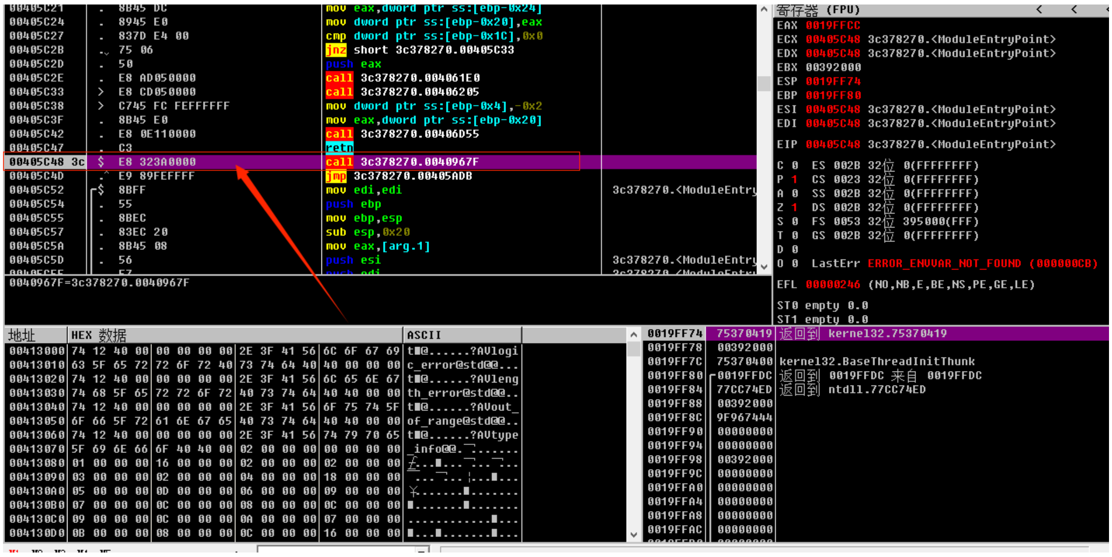
一、分析过程如下

通过 DIE 判断为加壳，开发语言为 c/c++



```
Detect It Easy 1.00
文件名: Jsers/Administrator/Downloads/样本/3c37827070f8d4eb726f59a0d4f2db0d8f1232ca
类型: PE    大小: 266752
EntryPoint: 00005c48          ImageBase: 00400000
NumberOfSections: 0003        SizeOfImage: 028e8000
compiler    Microsoft Visual C/C++(2010)[libcmt]
linker      Microsoft Linker(10.0)[EXE32]
100%    187 ms
```

程序入口点 5c48,使用编辑器为 vs 编辑器



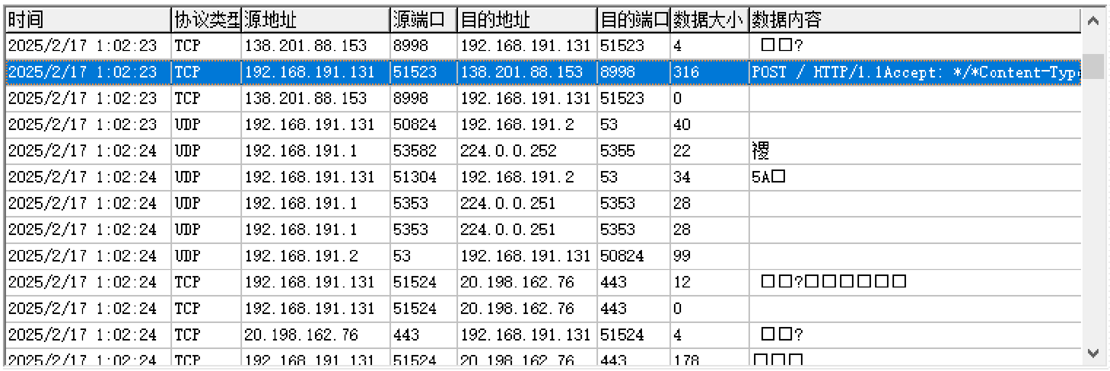
```
00405C24     .  8945 E0          mov dword ptr ss:[ebp-0x20],eax
00405C27     .  837D E4 00       cmp dword ptr ss:[ebp-0x1C],0x0
00405C2B     .. 75 06            jnz short 3c378270.00405C33
00405C2D     .  50               push eax
00405C2E     .  E8 AD050000      call 3c378270.004061E0
00405C33     >  E8 CD050000      call 3c378270.00406205
00405C38     >  C745 FC FEFFFFFF mov dword ptr ss:[ebp-0x4],-0x2
00405C3F     .  8B45 E0          mov eax,dword ptr ss:[ebp-0x20]
00405C42     .  E8 0E110000      call 3c378270.00406D55
00405C47     .  C3               retn
00405C48 3c  $  E8 323A0000      call 3c378270.0040967F
00405C4D     .  E9 89FEFFFF      jmp 3c378270.00405ADB
00405C52     ┌$ 8BFF             mov edi,edi                 3c378270.<ModuleEntry
00405C54     .  55               push ebp
00405C55     .  8BEC             mov ebp,esp
00405C57     .  83EC 20          sub esp,0x20
00405C5A     .  8B45 08          mov eax,[arg.1]
00405C5D     .  56               push esi                    3c378270.<ModuleEntry
0040967F=3c378270.0040967F
```

```
寄存器 (FPU)
EAX 0019FFCC
ECX 00405C48 3c378270.<ModuleEntryPoint>
EDX 00405C48 3c378270.<ModuleEntryPoint>
EBX 00392000
ESP 0019FF74
EBP 0019FF80
ESI 00405C48 3c378270.<ModuleEntryPoint>
EDI 00405C48 3c378270.<ModuleEntryPoint>

EIP 00405C48 3c378270.<ModuleEntryPoint>

C 0  ES 002B 32位 0(FFFFFFFF)
P 1  CS 0023 32位 0(FFFFFFFF)
A 0  SS 002B 32位 0(FFFFFFFF)
Z 1  DS 002B 32位 0(FFFFFFFF)
S 0  FS 0053 32位 395000(FFF)
T 0  GS 002B 32位 0(FFFFFFFF)
D 0
O 0  LastErr ERROR_ENVVAR_NOT_FOUND (000000CB)

EFL 00000246 (NO,NB,E,BE,NS,PE,GE,LE)

ST0 empty 0.0
ST1 empty 0.0
```

```
地址     HEX 数据                                         ASCII
00413000 74 12 40 00 00 00 00 00 2E 3F 41 56 6C 6F 67 69 t▮@......?AVlogi
00413010 63 5F 65 72 72 6F 72 40 73 74 64 40 40 00 00 00 c_error@std@@...
00413020 74 12 40 00 00 00 00 00 2E 3F 41 56 6C 65 6E 67 t▮@......?AVleng
00413030 74 68 5F 65 72 72 6F 72 40 73 74 64 40 40 00 00 th_error@std@@..
00413040 74 12 40 00 00 00 00 00 2E 3F 41 56 6F 75 74 5F t▮@......?AVout_
00413050 6F 66 5F 72 61 6E 67 65 40 73 74 64 40 40 00 00 of_range@std@@..
00413060 74 12 40 00 00 00 00 00 2E 3F 41 56 74 79 70 65 t▮@......?AVtype
00413070 5F 69 6E 66 6F 40 40 00 02 00 00 00 00 00 00 00 _info@@.........
00413080 01 00 00 00 16 00 00 00 02 00 00 00 02 00 00 00 ....▮...........
00413090 03 00 00 00 02 00 00 00 04 00 00 00 18 00 00 00 ............▮...
004130A0 05 00 00 00 0D 00 00 00 06 00 00 00 09 00 00 00 ........▮.......
004130B0 07 00 00 00 0C 00 00 00 08 00 00 00 0C 00 00 00 ▮.......▮.......
004130C0 09 00 00 00 0C 00 00 00 0A 00 00 00 07 00 00 00 ............▮...
004130D0 0B 00 00 00 08 00 00 00 0C 00 00 00 16 00 00 00 ▮...▮.......▮...
```

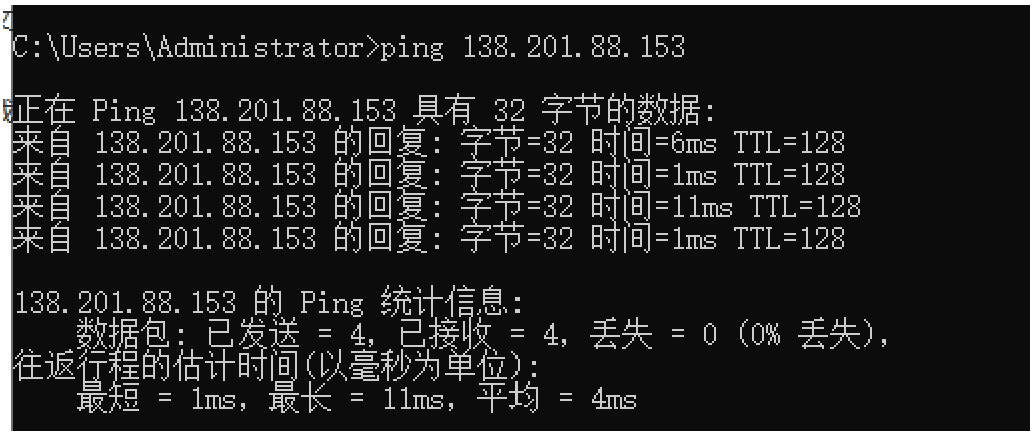
```
0019FF74  75370419 返回到 kernel32.75370419
0019FF78  00392000
0019FF7C  75370400 kernel32.BaseThreadInitThunk
0019FF80 ┌0019FFDC 返回到 0019FFDC 来自 0019FFDC
0019FF84 │77CC74ED 返回到 ntdll.77CC74ED
0019FF88 │00392000
0019FF8C │9F967444
0019FF90 │00000000
0019FF94 │00000000
0019FF98 │00392000
0019FF9C │00000000
0019FFA0 │00000000
0019FFA4 │00000000
0019FFA8 │00000000
0019FFAC │00000000
```

二、行为分析

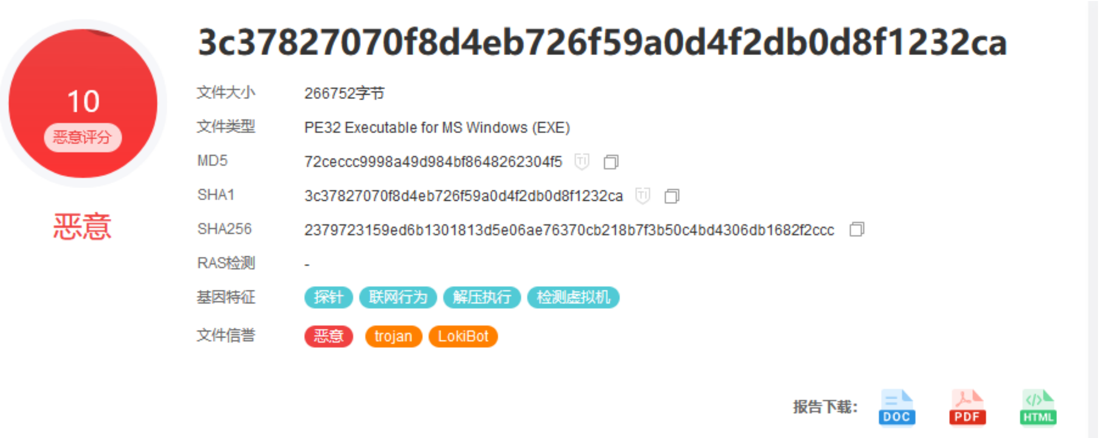


时间	协议类型	源地址	源端口	目的地址	目的端口	数据大小	数据内容
2025/2/17 1:02:23	TCP	138.201.88.153	8998	192.168.191.131	51523	4	□□?
2025/2/17 1:02:23	TCP	192.168.191.131	51523	138.201.88.153	8998	316	POST / HTTP/1.1Accept: */*Content-Typ
2025/2/17 1:02:23	TCP	138.201.88.153	8998	192.168.191.131	51523	0	
2025/2/17 1:02:23	UDP	192.168.191.131	50824	192.168.191.2	53	40	
2025/2/17 1:02:24	UDP	192.168.191.1	53582	224.0.0.252	5355	22	褸
2025/2/17 1:02:24	UDP	192.168.191.131	51304	192.168.191.2	53	34	5A□
2025/2/17 1:02:24	UDP	192.168.191.1	5353	224.0.0.251	5353	28	
2025/2/17 1:02:24	UDP	192.168.191.1	5353	224.0.0.251	5353	28	
2025/2/17 1:02:24	UDP	192.168.191.2	53	192.168.191.131	50824	99	
2025/2/17 1:02:24	TCP	192.168.191.131	51524	20.198.162.76	443	12	□□?□□□□□□□
2025/2/17 1:02:24	TCP	192.168.191.131	51524	20.198.162.76	443	0	
2025/2/17 1:02:24	TCP	20.198.162.76	443	192.168.191.131	51524	4	□□?
2025/2/17 1:02:24	TCP	192.168.191.131	51524	20.198.162.76	443	178	□□□

发现有一个恶意连接 ip 为 138 的 ip 地址

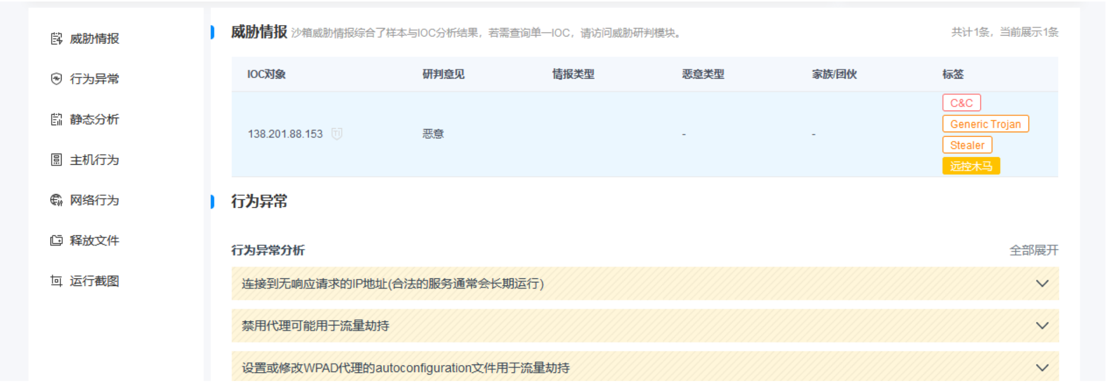


```
C:\Users\Administrator>ping 138.201.88.153

正在 Ping 138.201.88.153 具有 32 字节的数据:
来自 138.201.88.153 的回复: 字节=32 时间=6ms TTL=128
来自 138.201.88.153 的回复: 字节=32 时间=1ms TTL=128
来自 138.201.88.153 的回复: 字节=32 时间=11ms TTL=128
来自 138.201.88.153 的回复: 字节=32 时间=1ms TTL=128

138.201.88.153 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 1ms, 最长 = 11ms, 平均 = 4ms
```



3c37827070f8d4eb726f59a0d4f2db0d8f1232ca

文件大小	266752字节
文件类型	PE32 Executable for MS Windows (EXE)
MD5	72ceccc9998a49d984bf864826230 4f5
SHA1	3c37827070f8d4eb726f59a0d4f2db0d8f1232ca
SHA256	2379723159ed6b1301813d5e06ae76370cb218b7f3b50c4bd4306db1682f2ccc
RAS检测	-
基因特征	探针 联网行为 解压执行 检测虚拟机
文件信誉	恶意 trojan LokiBot



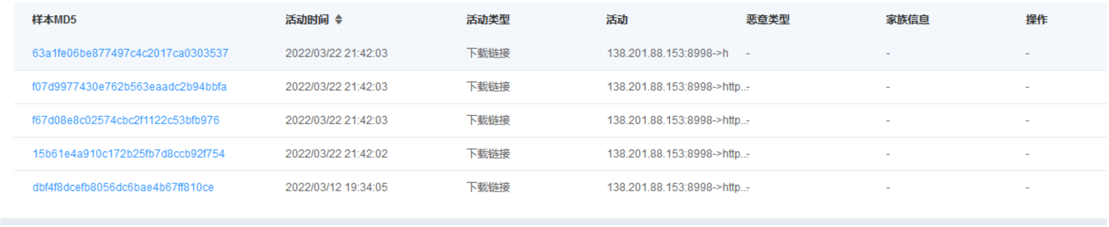
威胁情报 沙箱威胁情报综合了样本与IOC分析结果，若需查询单一IOC，请访问威胁研判模块。 共计1条，当前展示1条

IOC对象	研判意见	情报类型	恶意类型	家族/团伙	标签
138.201.88.153	恶意		-	-	C&C Generic Trojan Stealer 远控木马

行为异常

行为异常分析 全部展开

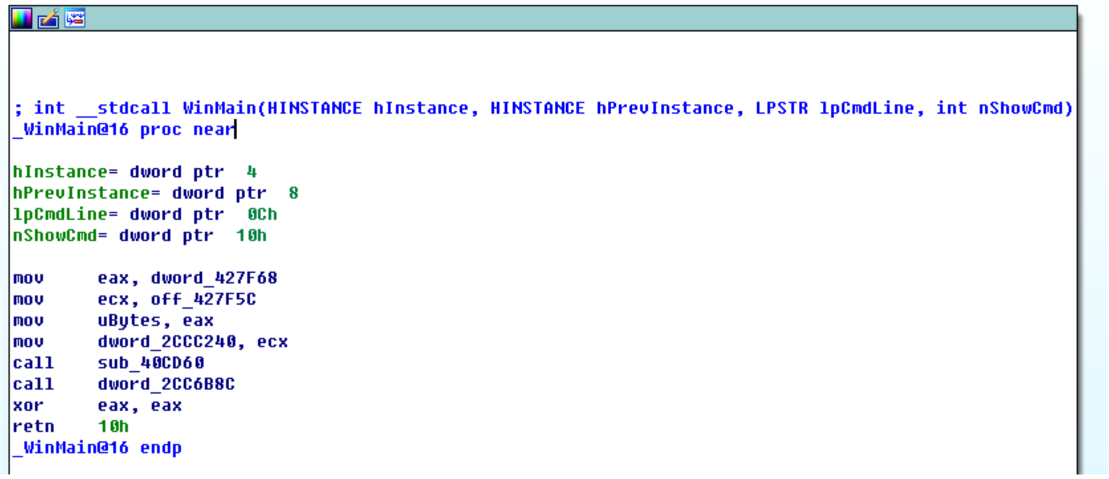
- 连接到无响应请求的IP地址(合法的服务通常会长期运行)
- 禁用代理可能用于流量劫持
- 设置或修改WPAD代理的autoconfiguration文件用于流量劫持



样本MD5	活动时间 ⇕	活动类型	活动	恶意类型	家族信息	操作
63a1fe06be877497c4c2017ca0303537	2022/03/22 21:42:03	下载链接	138.201.88.153:8998->h	-	-	-
f07d9977430e762b563eaadc2b94bbfa	2022/03/22 21:42:03	下载链接	138.201.88.153:8998->http..	-	-	-
f67d08e8c02574cbc2f1122c53bfb976	2022/03/22 21:42:03	下载链接	138.201.88.153:8998->http..	-	-	-
15b61e4a910c172b25fb7d8ccb92f754	2022/03/22 21:42:02	下载链接	138.201.88.153:8998->http..	-	-	-
dbf4f8dcefb8056dc6bae4b67ff810ce	2022/03/12 19:34:05	下载链接	138.201.88.153:8998->http..	-	-	-

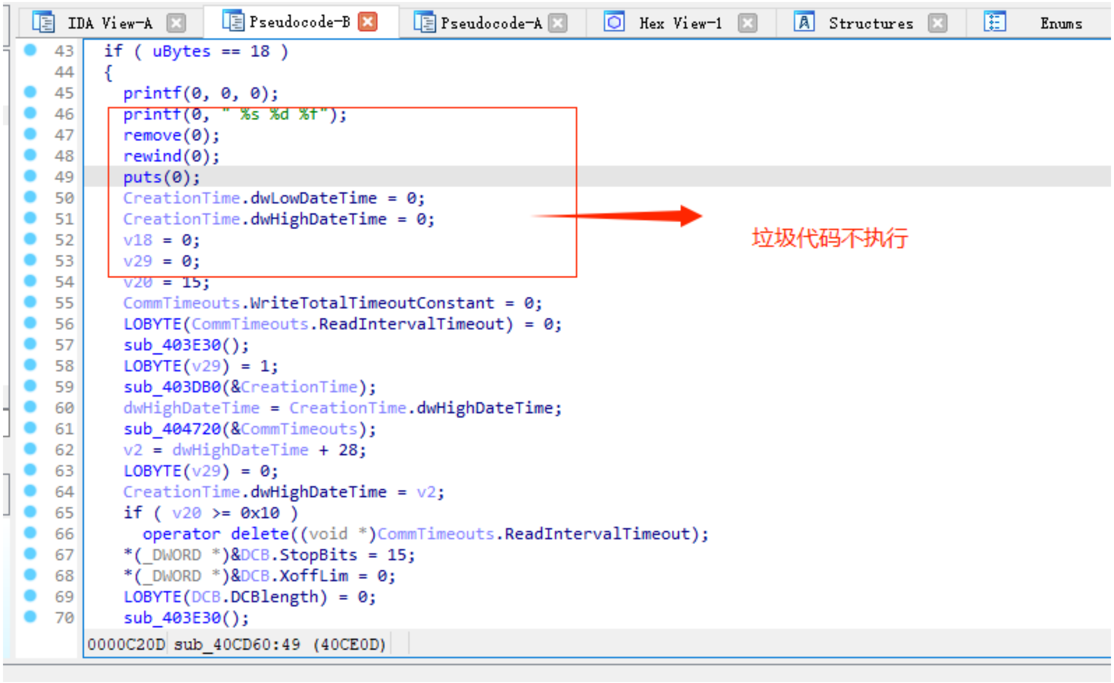
三、详细分析（静态+动态）

1）首先跟进主函数 winmain



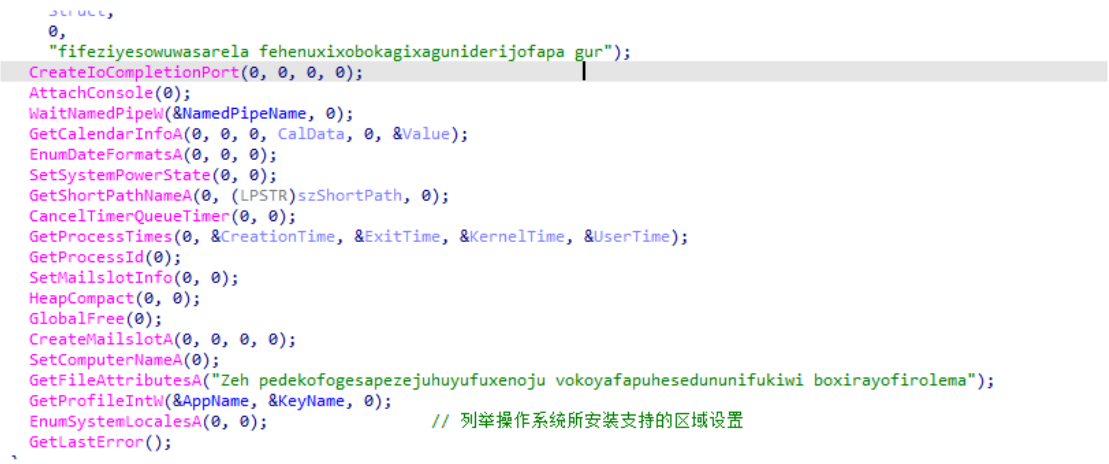
```
; int __stdcall WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int nShowCmd)
_WinMain@16 proc near

hInstance= dword ptr  4
hPrevInstance= dword ptr  8
lpCmdLine= dword ptr  0Ch
nShowCmd= dword ptr  10h

mov     eax, dword_427F68
mov     ecx, off_427F5C
mov     uBytes, eax
mov     dword_2CCC240, ecx
call    sub_40CD60
call    dword_2CC6B8C
xor     eax, eax
retn    10h
_WinMain@16 endp
```

跟进 40CD60

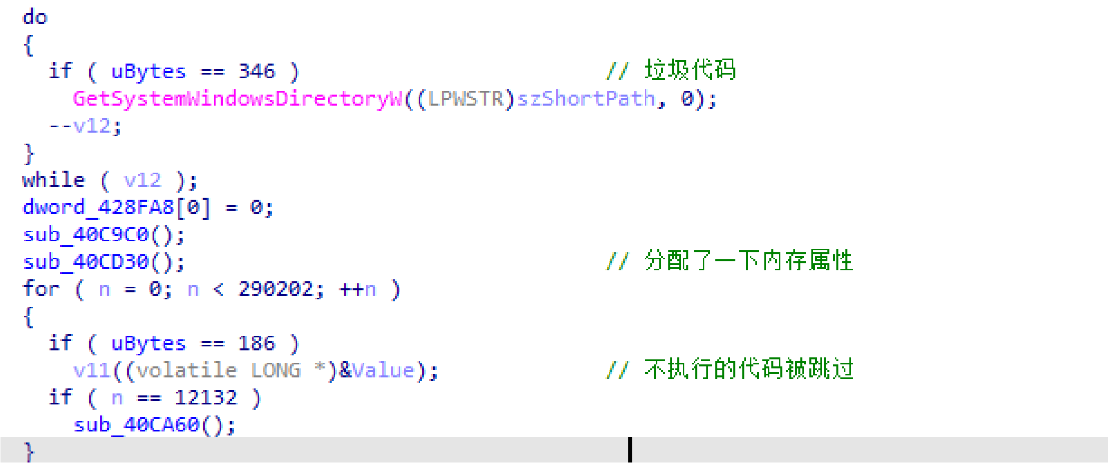


```
if ( uBytes == 18 )
{
  printf(0, 0, 0);
  printf(0, " %s %d %f");
  remove(0);
  rewind(0);
  puts(0);
  CreationTime.dwLowDateTime = 0;
  CreationTime.dwHighDateTime = 0;
  v18 = 0;
  v29 = 0;
  v20 = 15;
  CommTimeouts.WriteTotalTimeoutConstant = 0;
  LOBYTE(CommTimeouts.ReadIntervalTimeout) = 0;
  sub_403E30();
  LOBYTE(v29) = 1;
  sub_403DB0(&CreationTime);
  dwHighDateTime = CreationTime.dwHighDateTime;
  sub_404720(&CommTimeouts);
  v2 = dwHighDateTime + 28;
  LOBYTE(v29) = 0;
  CreationTime.dwHighDateTime = v2;
  if ( v20 >= 0x10 )
    operator delete((void *)CommTimeouts.ReadIntervalTimeout);
  *(_DWORD *)&DCB.StopBits = 15;
  *(_DWORD *)&DCB.XoffLim = 0;
  LOBYTE(DCB.DCBlength) = 0;
  sub_403E30();
0000C20D sub_40CD60:49 (40CE0D)
```

垃圾代码不执行

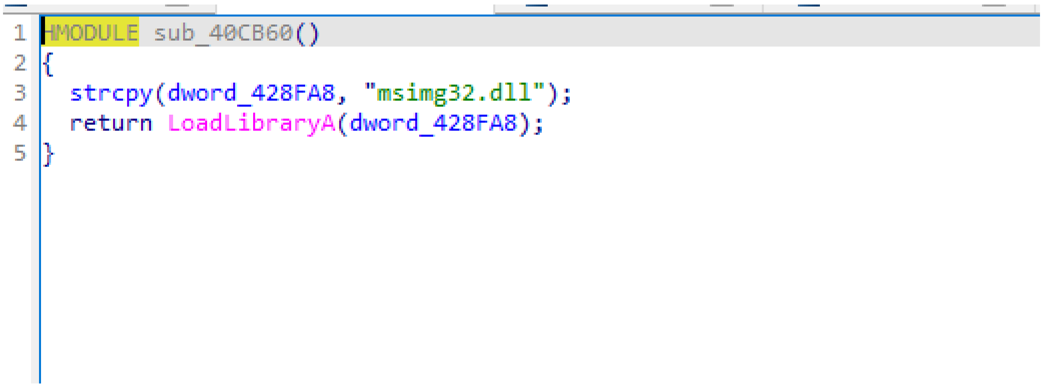


```
    struct,
    0,
    "fifeziyesowuwasarela fehenuxixobokagixaguniderijofapa gur");
  CreateIoCompletionPort(0, 0, 0, 0);
  AttachConsole(0);
  WaitNamedPipeW(&NamedPipeName, 0);
  GetCalendarInfoA(0, 0, 0, CalData, 0, &Value);
  EnumDateFormatsA(0, 0, 0);
  SetSystemPowerState(0, 0);
  GetShortPathNameA(0, (LPSTR)szShortPath, 0);
  CancelTimerQueueTimer(0, 0);
  GetProcessTimes(0, &CreationTime, &ExitTime, &KernelTime, &UserTime);
  GetProcessId(0);
  SetMailslotInfo(0, 0);
  HeapCompact(0, 0);
  GlobalFree(0);
  CreateMailslotA(0, 0, 0, 0);
  SetComputerNameA(0);
  GetFileAttributesA("Zeh pedekofogesapezejuhuyufuxenoju vokoyafapuhesedununifukiwi boxirayofirolema");
  GetProfileIntW(&AppName, &KeyName, 0);
  EnumSystemLocalesA(0, 0);                     // 列举操作系统所安装支持的区域设置
  GetLastError();
```



```
do
{
  if ( uBytes == 346 )                        // 垃圾代码
    GetSystemWindowsDirectoryW((LPWSTR)szShortPath, 0);
  --v12;
}
while ( v12 );
dword_428FA8[0] = 0;
sub_40C9C0();
sub_40CD30();                                 // 分配了一下内存属性
for ( n = 0; n < 290202; ++n )
{
  if ( uBytes == 186 )
    v11((volatile LONG *)&Value);             // 不执行的代码被跳过
  if ( n == 12132 )
    sub_40CA60();
}
```

动态加载 msing32.dll



```
HMODULE sub_40CB60()
{
  strcpy(dword_428FA8, "msimg32.dll");
  return LoadLibraryA(dword_428FA8);
}
```

四、总结

这是一个远控类型的病毒，截至到目前，该 c2 地址仍然在使用状态，具有监控屏幕，操作文件等功能