Analysis of W97M.Obfus.Generic

PREPARED BY

Argonyte

1. File Delivery Vector/General Analysis

W97M.Obfus.Generic is not explicitly malicious but has macros that will load functionality from Dynamically linked libraries (DLLs). It is usually used in accessing APIs that are usually not exposed by the VBA.

Malware was created in the year of 2006 and the document usually had one to three pages of content.

SHA1: de67fa75581f52dcc8262efc97b5b2748e8185dd798ebc2f85ff71fe07a27ead
Size: 328.2 kB
File and MIME Type: Word document (application/msword)
AV Detection Rate: 1/58

```
remnux@remnux:~/Downloads/de67fa75581f52dcc8262efc97b5b2748e8185dd798ebc2f85ff71fe07a27ead$ file de67fa75581f52dcc8262efc97b5b2748e8185dd798ebc2
f85ff71fe07a27ead.vx
de67fa75581f52dcc8262efc97b5b2748e8185dd798ebc2f85ff71fe07a27ead.vx: Composite Document File V2 Document, Little Endian, Os: Windows, Version 6.
1, Code page: 936, Title: 0, Author: cxf, Template: wgsjthzp.dot, Last Saved By: lyh(lyh), Revision Number: 2210, Name of Creating Application:
Microsoft Office Word, Total Editing Time: 2d+11:05:00, Last Printed: Fri May 30 03:08:00 2003, Create Time/Date: Sun Apr 30 08:31:00 2006, Last
 Saved Time/Date: Wed Jul 29 04:16:00 2020, Number of Pages: 1, Number of Words: 66, Number of Characters: 377, Security: 0
```

2. Exiftool Analysis

Using Exiftool for more in depth analysis about the Malware. It shows the details related to the malware like the creation date, modification date, file type, last modified by, etc.

```
File Modification Date/Time     : 2020:08:26 13:46:02-04:00
File Access Date/Time           : 2020:08:27 08:38:50-04:00
File Inode Change Date/Time     : 2020:08:27 08:38:38-04:00
File Permissions                : rw-r--r--
File Type                       : DOC
File Type Extension             : doc
MIME Type                       : application/msword
Identification                  : Word 8.0
Language Code                   : English (US)
Doc Flags                       : Has picture, 1Table, ExtChar, Far east
System                          : Windows
Word 97                         : No
Title                           : 0????????
Subject                         :
Author                          : cxf
Keywords                        :
Comments                        :
Template                        : wgsjthzp.dot
Last Modified By                : lyh(lyh)
Software                        : Microsoft Office Word
Create Date                     : 2006:05:31 07:31:00
Modify Date                     : 2020:07:29 03:16:00
Security                        : None
Company                         : sipo
Char Count With Spaces          : 442
App Version                     : 12.0000
Scale Crop                      : No
Links Up To Date                : No
Shared Doc                      : No
Hyperlinks Changed              : No
Code Page                       : Windows Simplified Chinese (PRC, Singapore)
XML Loaded                      : 0
Doc Guid                        : 16-08-19-16-37-40
Edition                         : 1.0.218/2016-05-20 16:02
Is Template                     : gwssiTemplate
Comp Obj User Type Len          : 35
Comp Obj User Type              : Microsoft Office Word 97-2003 ?j?
Last Printed                    : 2003:05:30 10:08:00Z
Revision Number                 : 2210
Total Edit Time                 : 2.5 days
Words                           : 66
Characters                      : 377
Pages                           : 1
Paragraphs                      : 1
```

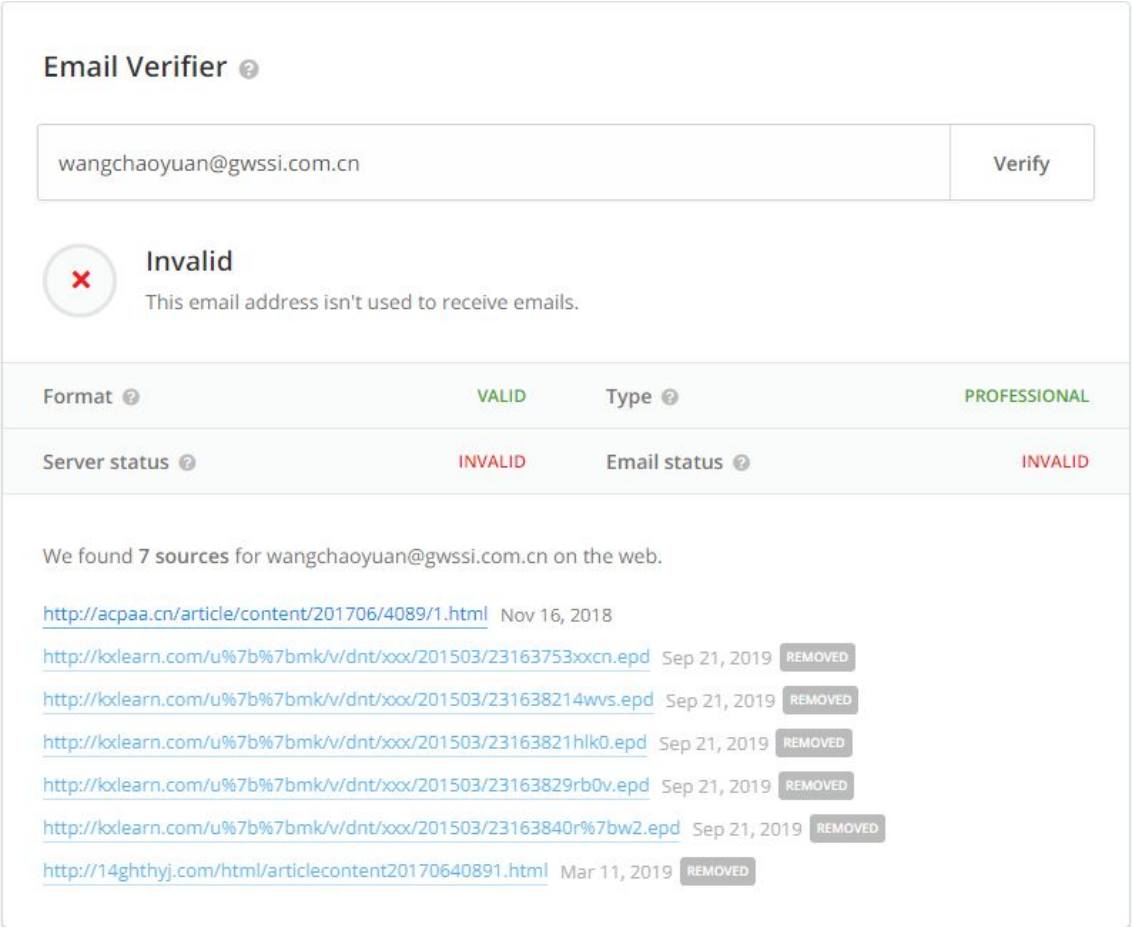
3. Strings Analysis

Initial Analysis after passing through strings reveal the document creator's email.

```
Name`
==========================================================================================
created by wangchaoyuan@gwssi.com.cn
date 2008-06-26
==========================================================================================
MSXML2.DOMDocument.4.0
```

Going over to the domain gwssi.com.cn gives us a website of a Chinese Infosec Company. So we can confirm that the document origination point is from China.

Checking the email that was shown in strings and verifying it through Hunter.io shows that the email isn't valid. Oddly, the email gets mentioned in a couple chinese articles and tenders for electronic equipment (only one exists as of now).



Doing WhoIs record check shows us no info about the domain itself as well.

Checking the DNS records of the site itself shows a bit more detail about the site itself.

gwssi.com.cn

HOST IP ADDRESS: 106.37.170.131

ASN:	AS4847
Continent:	Asia
Country:	China
Country Code:	CN
Country CF:	99
Region:	
State:	Beijing Shi
State Code:	
State CF:	98
DMA:	
MSA:	
City:	Beijing
Postal Code:	100032
Timezone:	Greenwich Mean Time
Area Code:	
City CF:	98
Latitude:	39.912289
Longitude:	116.365868
Map Location:	Click here

Powered by
Neustar IP Intelligence Logo

Checking the Site Status shows us even more details.

gwssi.com.cn

WEBSITE STATUS

```
    Status: Domain Registered and Website Active
  Response: 200
```

SSL INFORMATION

No Certificate found.

TITLE

```
??????????????
```

WEB SERVER

```
Apache-Coyote/1.1
```

META KEYWORDS

META DESCRIPTION

OTHER INFORMATION ABOUT GWSSI.COM.CN

```
 No. of Links on home page: 49
No. of Images on home page: 15
```

4. Olevba Analysis

Olevba is a script to parse OLE and OpenXML files such as MS Office documents (e.g. Word, Excel), to detect VBA Macros. Using this it shows us about the Malware is a table. Multiple macros and encrypted strings were detected and shown by it.

```
+----------+--------------------+--------------------------------------------+
|Type      |Keyword             |Description                                 |
+----------+--------------------+--------------------------------------------+
|AutoExec  |Document_Open       |Runs when the Word or Publisher document is |
|          |                    |opened                                      |
|AutoExec  |cmdClear_Click      |Runs when the file is opened and ActiveX    |
|          |                    |objects trigger events                      |
|AutoExec  |'生物材料样品保藏及存活信息_保藏单位代|Runs when the file is opened and ActiveX    |
|          |码_Change'           |objects trigger events                      |
|Suspicious|Open                |May open a file                             |
|Suspicious|Binary              |May read or write a binary file (if combined |
|          |                    |with Open)                                  |
|Suspicious|FileCopy            |May copy a file                             |
|Suspicious|CopyFile            |May copy a file                             |
|Suspicious|Kill                |May delete a file                           |
|Suspicious|vbNormal            |May run an executable file or a system      |
|          |                    |command                                     |
|Suspicious|CreateObject        |May create an OLE object                    |
|Suspicious|Windows             |May enumerate application windows (if       |
|          |                    |combined with Shell.Application object)     |
|Suspicious|FindWindow          |May enumerate application windows (if       |
|          |                    |combined with Shell.Application object)     |
|Suspicious|Lib                 |May run code from a DLL                     |
|Suspicious|Chr                 |May attempt to obfuscate specific strings   |
|          |                    |(use option --deobf to deobfuscate)         |
|Suspicious|SYSTEM              |May run an executable file or a system      |
|          |                    |command on a Mac (if combined with          |
|          |                    |libc.dylib)                                 |
|Suspicious|Hex Strings         |Hex-encoded strings were detected, may be   |
|          |                    |used to obfuscate strings (option --decode to|
|          |                    |see all)                                    |
|Suspicious|Base64 Strings      |Base64-encoded strings were detected, may be |
|          |                    |used to obfuscate strings (option --decode to|
|          |                    |see all)                                    |
```

5. Oledump.py Analysis

```
 1:       117 '\x01CompObj'
 2:       480 '\x05DocumentSummaryInformation'
 3:       448 '\x05SummaryInformation'
 4:     19098 '1Table'
 5:      4096 'Data'
 6:      1280 'Macros/PROJECT'
 7:        54 'Macros/PROJECTlk'
 8:       518 'Macros/PROJECTwm'
 9: M    1413 'Macros/VBA/NewXml'
10: M   14486 'Macros/VBA/ShowForm'
11: M    8155 'Macros/VBA/ThisDocument'
12:     16292 'Macros/VBA/_VBA_PROJECT'
13:     10861 'Macros/VBA/__SRP_0'
14:      1582 'Macros/VBA/__SRP_1'
15:       304 'Macros/VBA/__SRP_2'
16:      5476 'Macros/VBA/__SRP_3'
17:       741 'Macros/VBA/__SRP_4'
18:       768 'Macros/VBA/__SRP_5'
19:       481 'Macros/VBA/__SRP_6'
20:      2000 'Macros/VBA/__SRP_7'
21:      1694 'Macros/VBA/dir'
22: M    7401 'Macros/VBA/fileDialog'
23: M   31463 'Macros/VBA/frmAddPicture'
24: M    5500 'Macros/VBA/frmSup'
25: M   15490 'Macros/VBA/modAddPDF'
26: M    2468 'Macros/VBA/modChrProcess'
27: M    1311 'Macros/VBA/modDelFile'
28: M    5050 'Macros/VBA/modDya'
29: M    2508 'Macros/VBA/modInsertFile'
30: M    2142 'Macros/VBA/modInterFace'
31: M    6151 'Macros/VBA/modLoadTif'
32: M   14475 'Macros/VBA/modPic'
33: M   62960 'Macros/VBA/modPub'
34: M    1662 'Macros/VBA/modTreeInterFace'
35:       97 'Macros/frmAddPicture/\x01CompObj'
36:      334 'Macros/frmAddPicture/\x03VBFrame'
37:      185 'Macros/frmAddPicture/f'
38:      111 'Macros/frmAddPicture/i09/\x01CompObj'
39:      652 'Macros/frmAddPicture/i09/f'
40:      752 'Macros/frmAddPicture/i09/o'
41:      111 'Macros/frmAddPicture/i18/\x01CompObj'
42:      156 'Macros/frmAddPicture/i18/f'
43:    41688 'Macros/frmAddPicture/i18/o'
44:        0 'Macros/frmAddPicture/o'
45:       97 'Macros/frmSup/\x01CompObj'
46:      291 'Macros/frmSup/\x03VBFrame'
47:      429 'Macros/frmSup/f'
48:      444 'Macros/frmSup/o'
49:      116 'ObjectPool/_1657526526/\x01CompObj'
50:       44 'ObjectPool/_1657526526/\x03OCXNAME'
```

```
51:      6 'ObjectPool/_1657526526/\x03ObjInfo'
52:     84 'ObjectPool/_1657526526/contents'
53:    115 'ObjectPool/_1657526527/\x01CompObj'
54:     20 'ObjectPool/_1657526527/\x03OCXNAME'
55:      6 'ObjectPool/_1657526527/\x03ObjInfo'
56:    460 'ObjectPool/_1657526527/\x03PRINT'
57:    692 'ObjectPool/_1657526527/contents'
58:  12929 'WordDocument'
```

Oledump.py is a program to analyze OLE files (Compound File Binary Format). These files contain streams of data. oledump allows you to analyze these streams. Using this dump the streams into a txt file and analyze it. Oledump does not deobfuscate strings hence it shows certain encrypted strings.

```
Public Declare Function GetOpenFileName Lib "comdlg32.dll" Alias "GetOpenFileNameA" (pOpenfilename As OPENFILENAME) As Long
'1
'ÎÃ¼þ´ò¿º¶Ô»°¿ò
'º¯Êý:OpenFile
'²ÎÊý:WinHwnd   µ÷ÓÃ´Ëº¯ÊýµÄHWND
'    BoxLabel  ÉèÖÃ¶Ô»°¿òµÄ±êÇ©.
'    StartPath ÉèÖÃ³õÊ¼»¯Ä¬¾¶.
'    FilterStr ÎÃ¼þ¹ýÁÈ.
'    Flag      ±êÖ¾.(²Î¿¼MSDN)
'·µ»ØÖµ:String ÎÃ¼þÃû.
'Áý×ÓÉº
```

```
Sub InsertFile(sFilePath As String, sBkName As String)
'
' InsertFile Macro
' ºêÔÚ 2016-3-3 ÓÉ shenao Â¼ÖÆ
'
```

```
Attribute VB_Name = "modInsertFile"
'''''Sub InsertFile(sFilePath As String, sBkName As String)
'''''
''''' InsertFile Macro
''''' ºêÔÚ 2016-3-3 ÓÉ shenao Â¼ÖÆ
'''''
'''''    Dim fso As Scripting.FileSystemObject
'''''    Dim sFileName As String
'''''    Dim isp As InlineShape
'''''
'''''    UnlockDoc
'''''    Set fso = New FileSystemObject
'''''    fso.CopyFile sFilePath, fso.BuildPath(ThisDocument.Path, fso.GetFileName(sFilePath)), True
'''''
'''''    sFileName = fso.GetFileName(sFilePath)
'''''
'''''    Set isp = ThisDocument.Bookmarks("file_insert_mark").Range.InlineShapes.AddOLEObject(fileName:= _
'''''        sFilePath, LinkToFile:=False, _
'''''        DisplayAsIcon:=True, _
'''''        IconIndex:=0, IconLabel:=sFileName)
'''''
'''''    isp.Range.Bookmarks.Add sBkName
'''''    isp.AlternativeText = sFileName
'''''    Set fso = Nothing
'''''    LockDoc
'''''End Sub
```

```
'_ù¼Ýform¶ÖÏóµÄÄû³ÆªÍÄÚÈÝÉú³ÉxmlÍÄ½þ£¬´Ê¤ˆÊýÍª¨ÓÃº˜Êý
'''''Public Function createFormXMLTemplate(FileName As String) As Boolean
'''''    Dim nodLst As Object ' MSXML2.IXMLDOMNodeList
'''''    Dim nod As MSXML2.IXMLDOMNode
'''''    Dim cNod As Object ' IXMLDOMNode
'''''    Dim fld As FormField
'''''    Dim rootName As String
'''''    Dim id As String
'''''    Dim i As Integer
'''''    Dim xmlObj As MSXML2.DOMDocument40
'''''    On Error GoTo l
'''''    ThisDocument.Application.ScreenUpdating = False
'''''    Set xmlObj = CreateObject(XML_OBJ_NAME) ' New DOMDocument
'''''
'''''    'ÐÅ¿Ø¼þÐÞ,ÄÊ±Íí¼Ó
'''''    xmlObj.validateOnParse = False
'''''    xmlObj.resolveExternals = False
'''''
''''''            "<?xml-stylesheet type=" & """" & "text/xsl" & """" & " href=" & """" & "/dtdandxsl/showxml.xsl" & """" & "?>" & _"<?xml version=" & """" & "1.0" & """" & " encoding=" & """" & "UTF-8" & """" & "?>" & _
''''''
'''''    xmlstr = "<?xml version=" & """" & "1.0" & """" & " encoding=" & """" & "UTF-8" & """" & "?>" & _
'''''            "<!-- µç×ÓÉêÇë1.0 --><!DOCTYPE cn-application-body SYSTEM " & """" & "/dtdandxsl/cn-application-body-20080416.dtd" & """" & ">" & _
'''''            "<cn-application-body lang=" & """" & "zh" & """" & " country=" & """" & "CN" & """" & ">" & _
'''''            "<cn-drawings>" & _
'''''            "</cn-drawings>" & _
'''''            "</cn-application-body>"
'''''    xmlObj.loadXML xmlstr
'''''    For Each nod In xmlObj.ChildNodes
'''''        If nod.NodeType = NODE_ELEMENT Then
'''''            rootName = nod.baseName
'''''        End If
'''''    Next
'''''    Index2Name xmlObj.SelectSingleNode("//" & rootName)
'''''    Dim NodeID As Integer
'''''    Dim PicNum As String
'''''    Dim PicWidth As Single
'''''    Dim PicHeight As Single
'''''    Dim imgformat As String
'''''    Dim imgName As String
'''''    Dim rootNode As IXMLDOMNode
'''''
'''''    Set rootNode = xmlObj.SelectSingleNode("//cn-application-body/cn-drawings/")
'''''    Dim j As Integer
'''''    For j = 2 To ThisDocument.FormFields.count
'''''        Set fld = ThisDocument.FormFields(j)
'''''        NodeID = NodeID + 1
'''''        PicNum = NodeID
'''''        PicWidth = PointsToMillimeters(fld.Range.Bookmarks(1).Range.InlineShapes(1).Width)
'''''        PicHeight = PointsToMillimeters(fld.Range.Bookmarks(1).Range.InlineShapes(1).Height)
```

```
'==========================================================================================
'´ÓxmlÍÄ¼þÕÐ×°ÕØÊý¾Ý
'==========================================================================================
''''Public Sub loadXmlData(xmlFileName As String) 'Í¯¹ýxmlÊý¾ÝÏÄ¼þºÍÃ£°æÌÄ¼þ»ÔÎîÐ´ÏÄ¼þ,¼ÓÔØxmlÊý¾Ýµ½´°ÌåÇØ¼þ
'''''    Dim nodLst As MSXML2.IXMLDOMNodeList ' Object ' MSXML2.IXMLDOMNodeList
'''''    Dim nod As MSXML2.IXMLDOMNode
'''''    Dim NodeName As String
'''''    Dim nodeValue As String
'''''    Dim rootName As String
'''''    Dim fld As FormField ' Object
'''''    Dim XmlObj As New DOMDocument40
'''''    Dim childNum As Integer
'''''    Dim i As Integer
'''''    Dim viewName As String
'''''    Dim filePath As String
'''''    Dim PicID As String
'''''    Dim PicWi As Single
'''''    Dim PicHe As Single
'''''    Dim PicObj As InlineShape
'''''    Dim LoadUnSuccessed As Boolean
'''''    If Trim(xmlFileName) = "" Then Exit Sub
'''''    If Dir(xmlFileName) = "" Then
'''''        Exit Sub
'''''    End If
'''''    On Error GoTo l
'''''    If GetProperty(XMLFlag) = True Then
'''''        Exit Sub
'''''    Else
'''''        UnlockDoc
'''''        ThisDocument.Range.Delete
'''''    End If
'''''    ThisDocument.Application.ScreenUpdating = False
'''''    Set XmlObj = CreateObject(XML_OBJ_NAME) ' New DOMDocument
'''''    'ÐÂ¿Ø¼þÐÞ´ÆÉ±ÍÍ¾Ó
'''''    XmlObj.validateOnParse = False
'''''    XmlObj.resolveExternals = False
'''''    XmlObj.Load xmlFileName
'''''    Set nodLst = XmlObj.SelectNodes("//figure")
'''''    childNum = nodLst.Length
'''''    If childNum > 0 Then
'''''        For i = 1 To childNum
'''''            Set nod = nodLst.Item(i - 1)
'''''            viewName = getAttrib(nod, "figure-labels")
'''''            PicID = getAttrib(nod, "num")
'''''            Set nod = nod.ChildNodes(0)
'''''             FilePath = getAttrib(nod, "file") ÐèÒªÐÞ¸ÄµÃµØ·½
'''''            filePath = Mid(xmlFileName, 1, InStrRev(xmlFileName, "\")) & getAttrib(nod, "file")

'''''            filePath = Mid(xmlFileName, 1, InStrRev(xmlFileName, "\")) & getAttrib(nod, "file")
'''''            PicWi = getAttrib(nod, "wi")
'''''            PicHe = getAttrib(nod, "he")
'''''            If Dir(filePath) <> "" Then
'''''                Set PicObj = addPicture(viewName, filePath)
'''''                PicObj.Width = MillimetersToPoints(PicWi)
'''''                PicObj.Height = MillimetersToPoints(PicHe)
'''''            Else
'''''                With ThisDocument
'''''                    Dim tmpPos As Long
'''''                    tmpPos = .Range.End - 1
'''''                    .Range(tmpPos, tmpPos + 1).Text = vbCrLf + "¼ÓÔØ´ÍíóÉºÍ¼Æ¬ÎÄ¼þ " & """" & filePath & """" & " Î´ÖÒµ½£¡" + vbCrLf  ', vbCritical, "´ÍíóíáÊ¾"
'''''                    .Range(tmpPos, .Range.End).Font.Color = wdColorRed
'''''                    LoadUnSuccessed = True
'''''                End With
''''''                Exit For
'''''            End If
'''''        Next i
'''''    End If
'''''    If LoadUnSuccessed = False Then
'''''        SetPropertyValue XMLFlag, True
'''''    End If
'''''    ThisDocument.Range(0, 0).Select
'''''    ThisDocument.Application.ScreenUpdating = True
'''''    LockDoc
'''''    Exit Sub
''''''l:
'''''    Err.Clear
'''''    Resume Next
''''''End Sub
```

```
"""        PicHeight = PointsToMillimeters(fld.Range.Bookmarks(1).Range.InlineShapes(1).Height)
"""        imgName = fld.Range.Bookmarks(1).Range.InlineShapes(1).LinkFormat.SourceName
"""        imgformat = LCase(Mid(imgName, InStrRev(imgName, ".") + 1))
"""
"""        Set nod = xmlObj.SelectSingleNode("//cn-application-body/cn-drawings")
"""        Set nod = nod.appendChild(createNode(xmlObj, "figure", ""))
"""        SetNodeAttribute xmlObj, nod, "id", "f" & Trim(CStr(NodeID))
"""        SetNodeAttribute xmlObj, nod, "num", Trim(CStr(NodeID))
"""        SetNodeAttribute xmlObj, nod, "figure-labels", Replace(GetPureResult(fld), Chr(13), "")
"""
"""        Set nod = nod.appendChild(createNode(xmlObj, "img", ""))
"""        SetNodeAttribute xmlObj, nod, "id", "if" & Trim(CStr(NodeID))
"""        SetNodeAttribute xmlObj, nod, "file", imgName
"""        SetNodeAttribute xmlObj, nod, "wi", Trim(CStr(PicWidth))
"""        SetNodeAttribute xmlObj, nod, "he", Trim(CStr(PicHeight))
"""        SetNodeAttribute xmlObj, nod, "img-content", "drawing"
"""        SetNodeAttribute xmlObj, nod, "img-format", imgformat
"""        SetNodeAttribute xmlObj, nod, "orientation", "portrait"
"""        SetNodeAttribute xmlObj, nod, "inline", "no"
"""      Next
"""      formatNodeIndex xmlObj.SelectSingleNode("//" & rootName)
"""      xmlObj.Save FileName '±£´æxmlÎÄ¼þ
"""      createFormXMLTemplate = True
"""      ThisDocument.Application.ScreenUpdating = True
"""      Exit Function
"""l:
"""      Err.Clear
"""      Resume Next
"""End Function 'createFormXML "c:\a.xml","110101·¢Ã÷×¨ÀûÇëÇóÊé.doc"
```

Using [Hybrid Analysis](#) for final analysis of the malware. It also shows Mitre's Att&ck Detection table.

MITRE ATT&CK™ Techniques Detection

Minimal

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
	User Execution 2 1	Hooking 3	Hooking 3	Modify Registry 1	Hooking 3	Application Window Discovery 1		Email Collection 1			
		Office Application Startup 1 1	Process Injection 1	Process Injection 1							