THEMED PHISHING

A Deep Dive into a Multifaceted COVID-19 Phishing Campaign

Robert Dean

Marshall University | April 22, 2020

During the event of any major tragedy, hackers have exploited human emotion and used themed phishing campaigns in their favor to profit from innocent people. Phishing attacks date back to the 1990s when hacker groups attempted to phish AOL users for their login credentials. Shortly after 9/11 attackers leveraged phishing attacks against US citizens to conduct fake ‘ID Checks’. December 2019 marks the beginning of the COVID-19 pandemic, a tragedy that has affected the entire world. Malicious actors took advantage of the crisis that was unraveling: many attackers utilized email, fake websites, and COVID-19 themed malware to infect users and steal information. Attackers have impersonated trusted organizations such as the World Health Organization and CDC, crafted COVID-19 vaccine websites and email campaigns, and created illegitimate COVID-19 mobile applications to manipulate the world’s population.

The definition of phishing, according to SANS, is “a type of attack that uses email or a messaging service to fool you into taking an action you should not take, such as clicking on a malicious link, sharing your password, or opening an infected email attachment” (Dudley 2018). Phishing attacks take on many forms. Sometimes, they may just be as simple as pretending to be trusted party (the World Health Organization, for example). Phishing can be used to trick users into providing login information or personally identifiable information (PII). This phishing tactic is referred to as social engineering, where the “hoax e-mails used in phishing schemes allege to [originate] from a trusted entity” (Elledge, 2007, p. 9). Attackers attempt to gain the victim’s trust to obtain sensitive information. Phishing attacks can also be used to distribute malware. Malware can take on many forms: for example, a ‘Trojan’ is a type of malware that appears as a legitimate application but conducts malicious activities. ‘Formbook’ malware is used to steal user credentials and form information from the user’s device. Depending on the attacker’s motive, many different types of malware may be distributed.

During the COVID-19 pandemic, hackers have used many platforms to steal personal data. In one case, a mobile Android application was created that presented a map of COVID-19 cases across the U.S. While this application appeared as legitimate, it was used to maliciously to encrypt the victim's data, "[requesting] $100 in bitcoin in 48 hours on the ransom note" (Saleh, 2020). This same type of COVID-19 application was implemented on Windows machines. The Windows version had "[a] real working interactive Coronavirus data map and a payload" that could infect targeted machines with malware (Krebs, 2020). In another example, attackers began an email campaign that advertised free COVID-19 tests to individuals. These emails contained a Microsoft Word document containing a 'request form' for each COVID-19 test. Once executed, the Word Document installs Trickbot malware. Trickbot is a type of malware used "to drop additional malware like ransomware, VNC clients and remote access malware" (Muncaster 2020), also known as a 'dropper'. This effectively grants an attacker full control of the compromised machine.

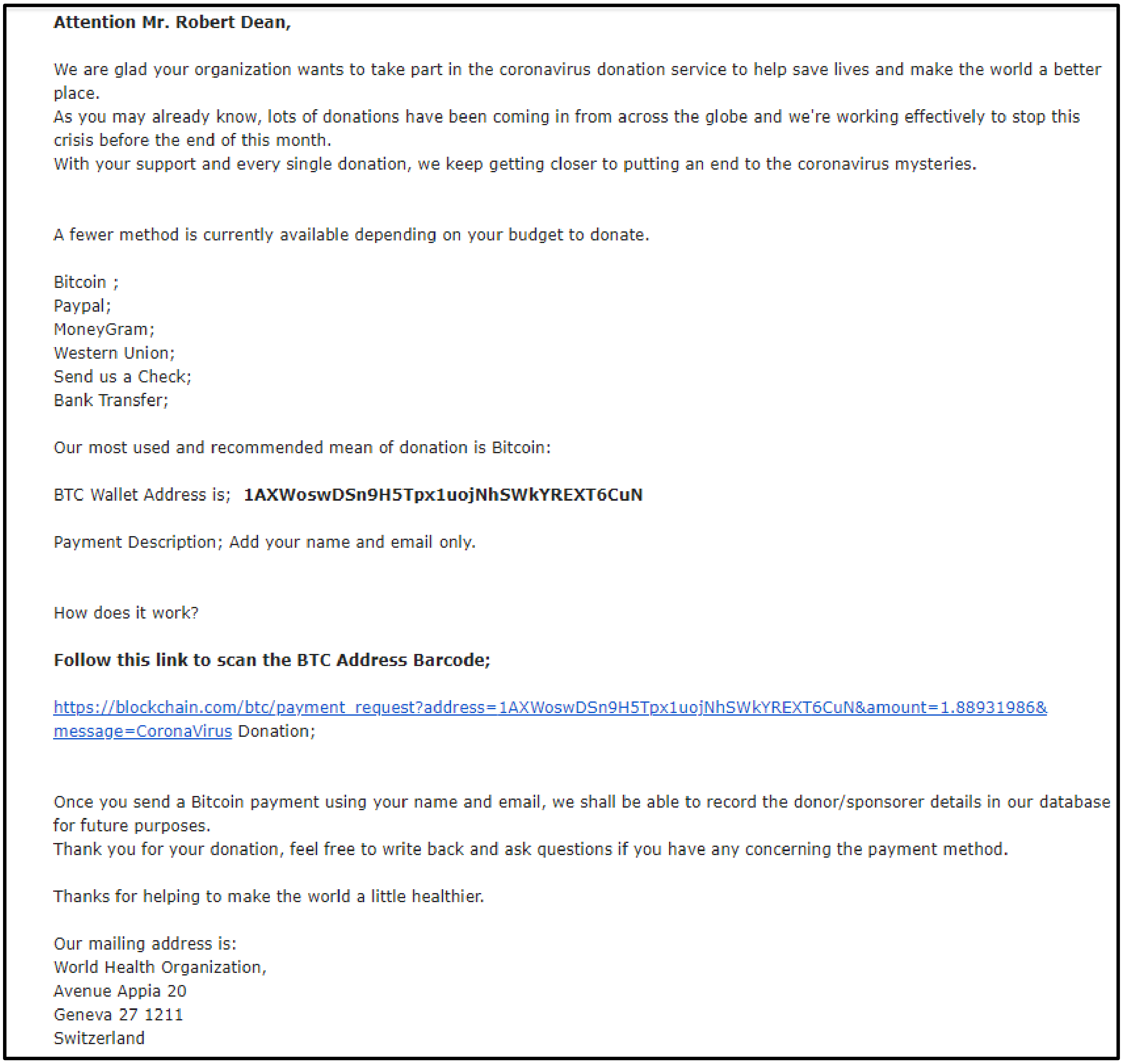
The purpose of this document will be to dive into a legitimate COVID-19 phishing attack that was used against the U.S. population. This attack combined malicious emails, a malicious website, and malware named 'GuLoader' that exfiltrates user data. The attack took place on Tuesday, March 17th at 5:18 AM. The malware discovered in this attack was recently named 'GuLoader' and was first discovered in March of 2020. This malware utilizes cloud shares to download encrypted payloads, which are then executed to conduct malicious activities on the compromised machine. In addition to exploring the capabilities of GuLoader, the phishing emails and website will be analyzed. This will provide an understanding of how attackers utilize many areas of technology to steal information.

Attack Storyline

This attack began with a series of malicious emails involving a COVID-19 ‘relief fund’ created by the World Health Organization. Like many phishing attacks, the attacker attempts to exploit human emotion and provide a sense of urgency so that the victim will act quickly. In this example, the attacker sends multiple emails impersonating the World Health Organization. The first email contains a simple request for donations to for the COVID-19 relief fund (**Figure 1)**. The second email is a follow up, providing updates and an e-book relating to the relief fund (**Figure 4**). A third email is sent, containing an illegitimate CDC website (**Figure 5**). The malicious actor leveraged multiple advanced phishing techniques in this attack.

Phishing Email One:

An initial email was received on Tuesday, March 17th at 5:18 AM from the sender corona-virus@caramail.com:



Attention Mr. Robert Dean,

We are glad your organization wants to take part in the coronavirus donation service to help save lives and make the world a better place.
As you may already know, lots of donations have been coming in from across the globe and we're working effectively to stop this crisis before the end of this month.
With your support and every single donation, we keep getting closer to putting an end to the coronavirus mysteries.

A fewer method is currently available depending on your budget to donate.

Bitcoin ;
Paypal;
MoneyGram;
Western Union;
Send us a Check;
Bank Transfer;

Our most used and recommended mean of donation is Bitcoin:

BTC Wallet Address is; **1AXWoswDSn9H5Tpx1uojNhSWkYREXT6CuN**

Payment Description; Add your name and email only.

How does it work?

Follow this link to scan the BTC Address Barcode;

https://blockchain.com/btc/payment_request?address=1AXWoswDSn9H5Tpx1uojNhSWkYREXT6CuN&amount=1.88931986&message=CoronaVirus Donation;

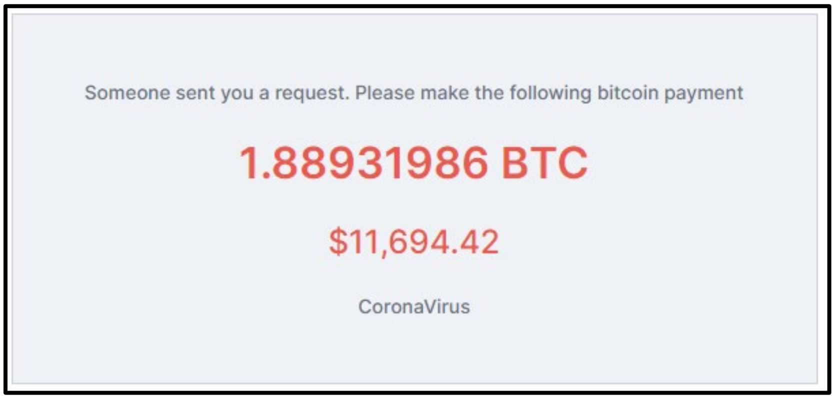
Once you send a Bitcoin payment using your name and email, we shall be able to record the donor/sponsorer details in our database for future purposes.
Thank you for your donation, feel free to write back and ask questions if you have any concerning the payment method.

Thanks for helping to make the world a little healthier.

Our mailing address is:
World Health Organization,
Avenue Appia 20
Geneva 27 1211
Switzerland

Figure 1: Initial Scam / Phishing Email sent by corona-virus@caramail.com

This email poses as the World Health Organization, requesting for donations to assist in combatting the COVID-19 pandemic. The attacker claims that if the recipient sends over a donation, it will help save lives and help end the COVID-19 crisis. They provide the legitimate mailing address of the World Health Organization in the end of the email. The attacker recommends sending donations via Bitcoin, as that is the "most used and recommended mean of donation". Bitcoin is also a cryptocurrency that is traded with anonymity in mind: the scammer's bitcoin wallet address provided in this email can't be traced back to him/her. The identity of the address also can't be confirmed to belong to the World Health Organization either. Visiting the link provided displays the initial request made by the attacker, including the specific amount that they requested:



Someone sent you a request. Please make the following bitcoin payment

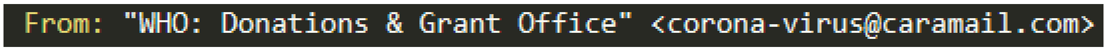
1.88931986 BTC

$11,694.42

CoronaVirus

Figure 2: blockchain.com link provided by the attacker

Observing the email header data, the attacker included "WHO: Donations & Grant Office" in the 'From' line in this email, attempting to impersonate the World Health Organization (**Figure 3)**.



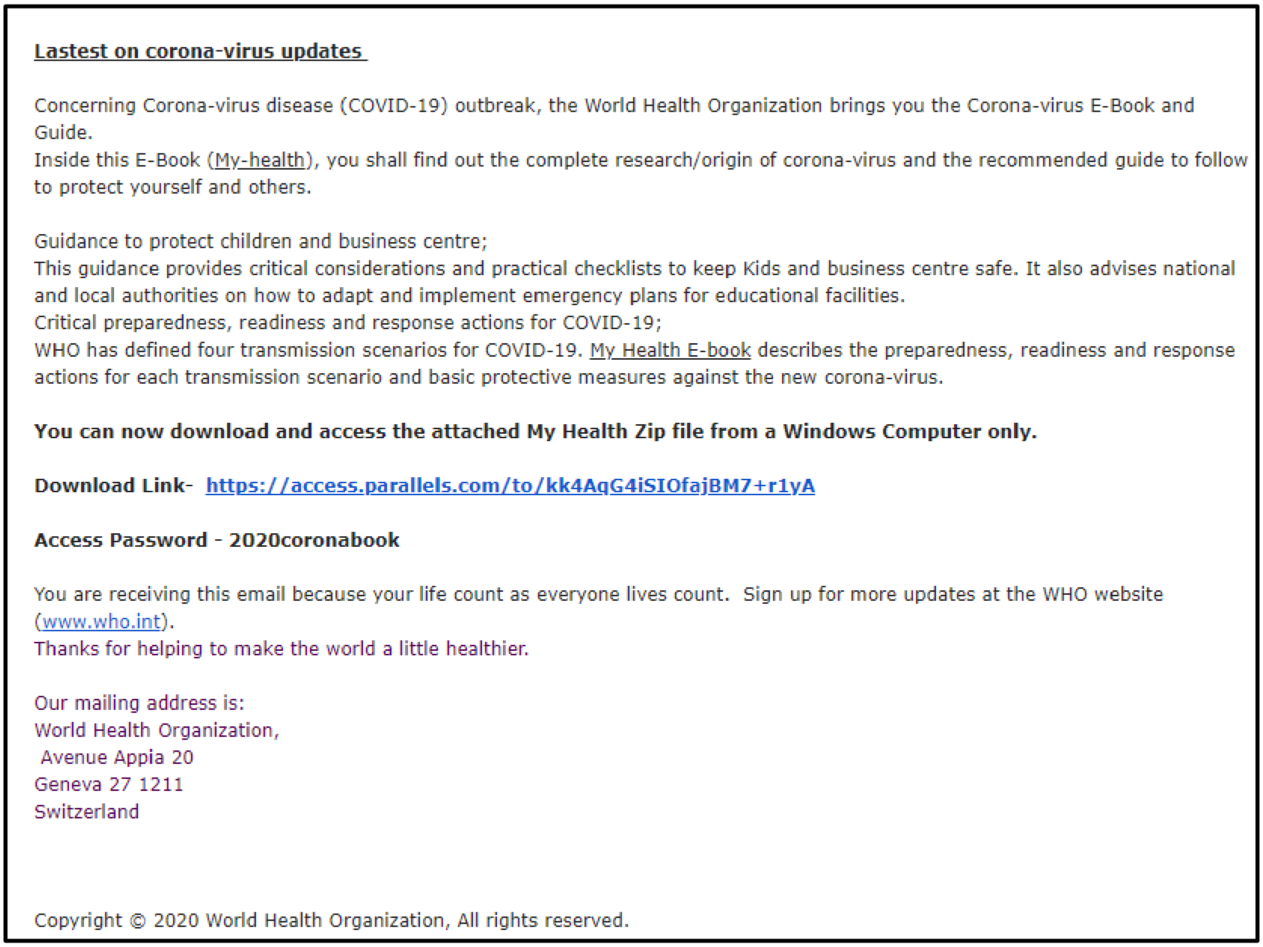
```
From: "WHO: Donations & Grant Office" <corona-virus@caramail.com>
```

Figure 3: 'From' line included in email header

Searching the web for the email "corona-virus@caramail.com" reveals its use in a phishing campaign that began in early March 2020. Attackers had previously utilized this email address to pose as the World Health Organization, attempting to distribute an "executable called MyHealth.exe" (Abrams, 2020). In addition to this malicious history, the domain "caramail.com" is a personal email service where this email was likely created.

Phishing Email Two:

The second part of this phishing campaign involves the delivery of malware. The attacker again, impersonates the World Health Organization using similar tactics mentioned previously. They introduce a ‘new’ COVID-19 e-book, named “My-Health”. The attacker claims the document provides “critical considerations and practical checklists to keep Kids and business centre safe”. It also claims to provide information about emergency plans and educational facilities.



Lastest on corona-virus updates

Concerning Corona-virus disease (COVID-19) outbreak, the World Health Organization brings you the Corona-virus E-Book and Guide.
Inside this E-Book (My-health), you shall find out the complete research/origin of corona-virus and the recommended guide to follow to protect yourself and others.

Guidance to protect children and business centre;
This guidance provides critical considerations and practical checklists to keep Kids and business centre safe. It also advises national and local authorities on how to adapt and implement emergency plans for educational facilities.
Critical preparedness, readiness and response actions for COVID-19;
WHO has defined four transmission scenarios for COVID-19. My Health E-book describes the preparedness, readiness and response actions for each transmission scenario and basic protective measures against the new corona-virus.

You can now download and access the attached My Health Zip file from a Windows Computer only.

Download Link- https://access.parallels.com/to/kk4AqG4iSIOfajBM7+r1yA

Access Password - 2020coronabook

You are receiving this email because your life count as everyone lives count. Sign up for more updates at the WHO website (www.who.int).
Thanks for helping to make the world a little healthier.

Our mailing address is:
World Health Organization,
Avenue Appia 20
Geneva 27 1211
Switzerland

Copyright © 2020 World Health Organization, All rights reserved.

Figure 4: Second Email in Phishing Campaign

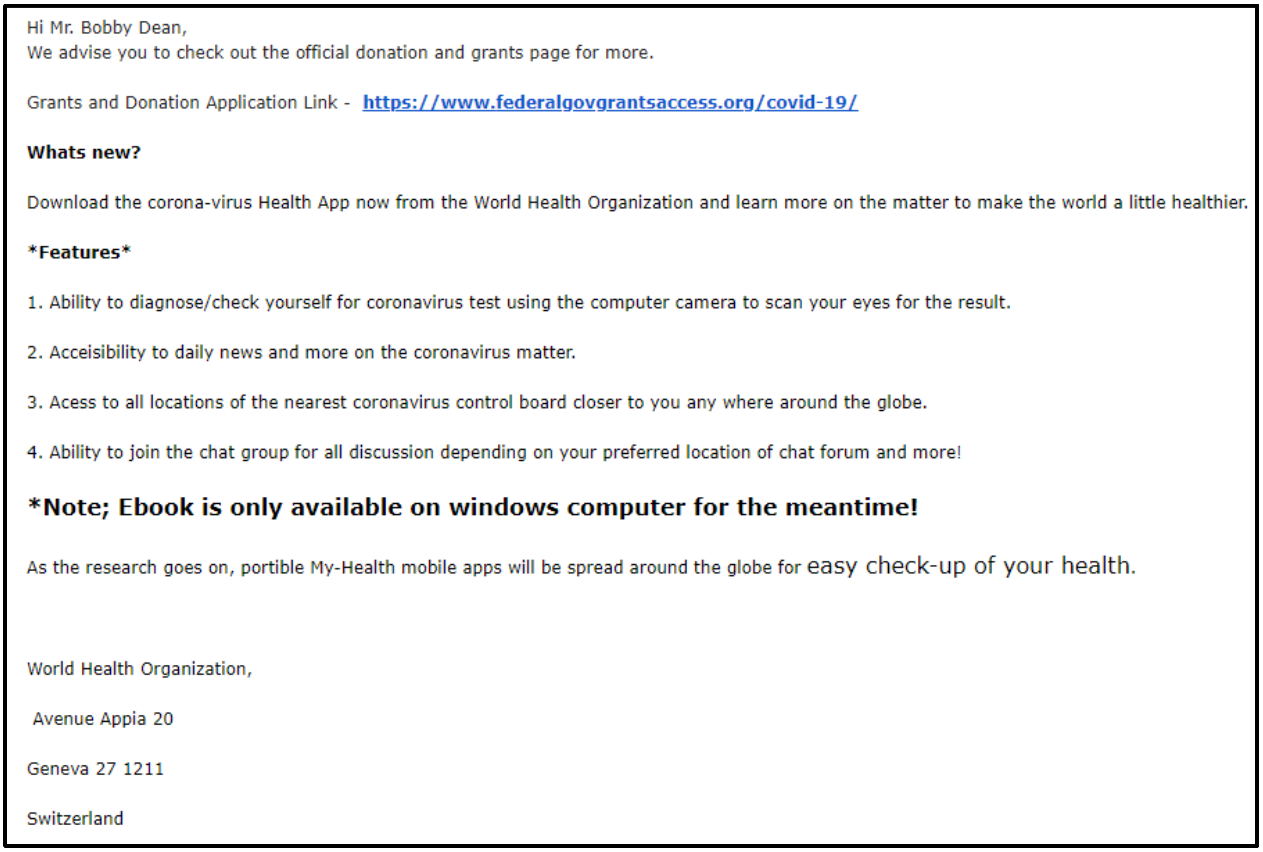
Provided in the email is a link to download the My-Health E-Book. It is hosted on https://access.parallels[.]com/. This website allows users to host files that are password protected. This file, named “Health-Ebook.zip”, will be analyzed in the **malware analysis** section.

Phishing Email Three:

A third email was received containing a URL to a ‘Grants and Donation Application” page:

https://www.federalgovgrantsaccess[.]org/covid-19/.

The attacker encourages the victim to download the “Corona-Virus Health App” from the World Health Organization (referring to the previous email).



Hi Mr. Bobby Dean,
We advise you to check out the official donation and grants page for more.

Grants and Donation Application Link - https://www.federalgovgrantsaccess.org/covid-19/

Whats new?

Download the corona-virus Health App now from the World Health Organization and learn more on the matter to make the world a little healthier.

Features

1. Ability to diagnose/check yourself for coronavirus test using the computer camera to scan your eyes for the result.

2. Acceisibility to daily news and more on the coronavirus matter.

3. Acess to all locations of the nearest coronavirus control board closer to you any where around the globe.

4. Ability to join the chat group for all discussion depending on your preferred location of chat forum and more!

***Note; Ebook is only available on windows computer for the meantime!**

As the research goes on, portible My-Health mobile apps will be spread around the globe for easy check-up of your health.

World Health Organization,

Avenue Appia 20

Geneva 27 1211

Switzerland

Figure 5: Email 3 from corona-virus@caramail.com

Upon analyzing email three, the spelling errors can be immediately noticed. These are the first indicator that the email is phony. One of the ‘features’ of the application is unrealistic: scanning a person’s eyes to provide a diagnosis for COVID-19. The attacker states that the “Ebook is only available on windows” devices. This malware, mentioned later, is used to target Windows machines. Lastly, the attacker prompts the user to go to visit their Grants and Donation Application Link: https://www.federalgovgrantsaccess[.]org/covid-19/. An analysis of this website provided will be included in the **Phishing Website** section.

Phishing Website

This section will provide an in-depth analysis of the URL provided in email 3: https://www.federalgovgrantsaccess[.]org/covid-19/

This website does not have a malicious reputation on VirusTotal. Upon visiting the website, it seems poorly constructed and does not appear as legitimate. The title is "WORLD HEALTH ORGANIZATION" and the website contains the World Health Organization logo in various places. It features 5 different pages: Home, Apply, Newsroom, Track Your Grant, and Contacts (**Figure 6**).

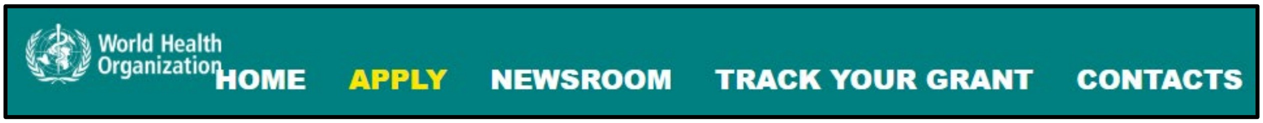


Figure 6: Pages included on Federalgovgrantaccess[.]org/covid-19/

Its homepage includes information on COVID-19, the Grand and Cooperative Agreement Act, and information regarding COVID-19 grants. The contact email at the bottom of the page is covid-19@federalgovgrantaccess.org and the phone number field is left blank (**Figure 7**).

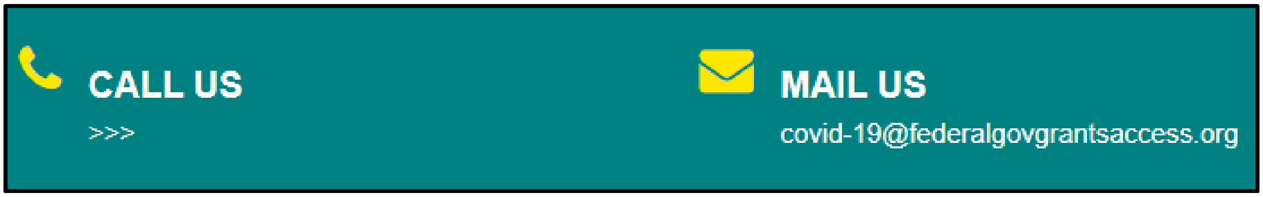
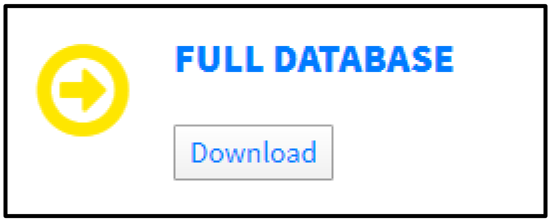


Figure 7: Contact Information for Federalgovgrantsaccess.org

The bottom of the homepage displays the text "Packet Pass © 2020". OSINT research of this company produces no results. The 'Apply' page provides an application form for those wishing to donate. It includes these fields: Full Legal Name, Email Address, Mobile Number, Residential

Address, Date of Birth, Marital Status, Nationality, Next of Kin, Own a Property, Grant/Donation Amount, Occupation, Monthly Income, and ID/Passport Number.

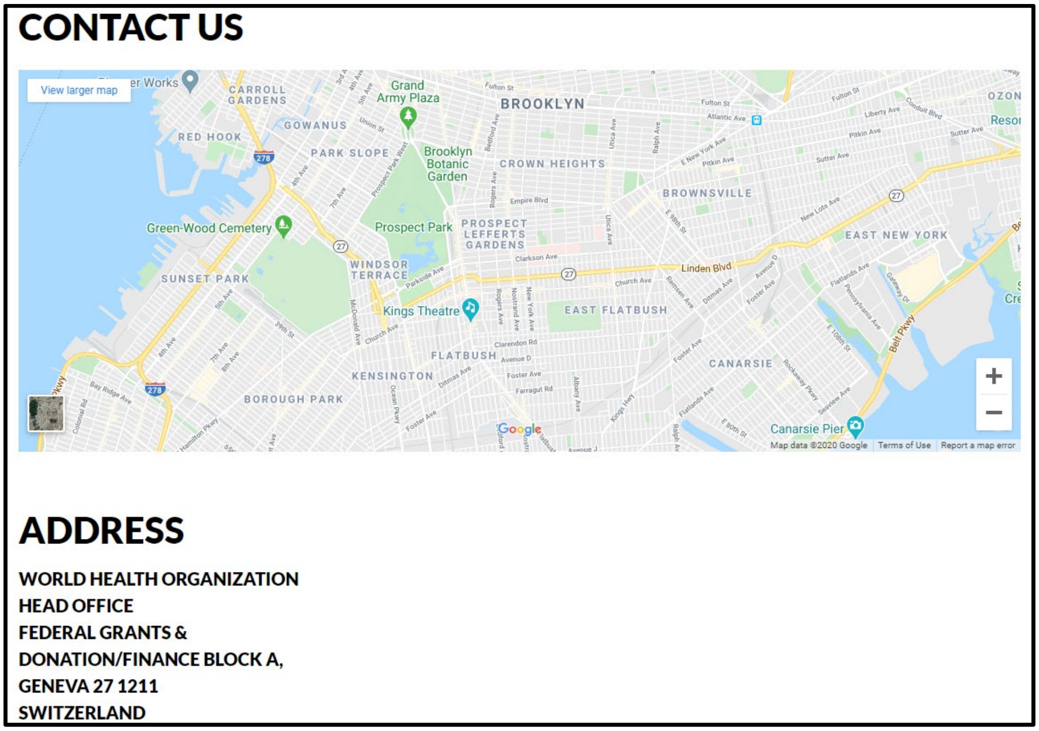
This page contains links to Twitter, Facebook, and even a 'Subscribe' button: all of these links are broken. The 'Newsroom' page contains a download link to a "Full Database" file. The link brings the user to the following page: https://www.federalgovgrantsaccess.org/covid-19/img-stk/COVID-19-Database-Files-2020.03.02.zip (**Figure 8**).



FULL DATABASE

Figure 8: Full Database Download Link

Unfortunately, the page presents an HTTP 404 (Not Found) when attempting to download the file. On the 'Contact' page, the attacker provides the legitimate address to the World Health Organization. The map on this page does not show the correct location, but rather an overview of Brooklyn, New York.



CONTACT US

ADDRESS

WORLD HEALTH ORGANIZATION
HEAD OFFICE
FEDERAL GRANTS &
DONATION/FINANCE BLOCK A,
GENEVA 27 1211
SWITZERLAND

Figure 9: Contact Page containing map of Brooklyn, NY

The flaws associated with this webpage give the appearance that it is a phishing site. A majority of the links are broken and there is no evidence providing that this website belongs to a legitimate government organization. Two weeks after this analysis was done, the website appears to be down, showing nothing but a blank page. Lastly, the email address ‘covid-19@caramail.com’ is listed on this site, tying this URL to the attacker.

Malware Analysis

The name of the file sent by the attacker is “Health-Ebook.exe”. This is stored inside a .zip folder named Health-Ebook.zip. This malware was analyzed in Windows 8, Windows 10, and Kali Linux virtual machines using VMWare Workstation 2020.1.

Using VirusTotal.com, the hash values are returned:

File Hashes

MD5: 93fba794dcb6996185f8e93062c12cd4

SHA-1: db73126ee8583999b121159e70e634ca23fd012d

SHA-256:
1e6bc511824f07c5107cb4a5075a811eb1d28f2916630bf7db1bb5c1649b0e7d

Virustotal.com also confirms the current and previous filenames:

File Names

Health-Ebook.exe

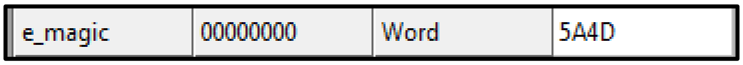
Cerithiumun

Cerithiumun.exe

Lastly, VirusTotal recognizes this file as malicious on 46/72 antivirus systems.

Static Analysis

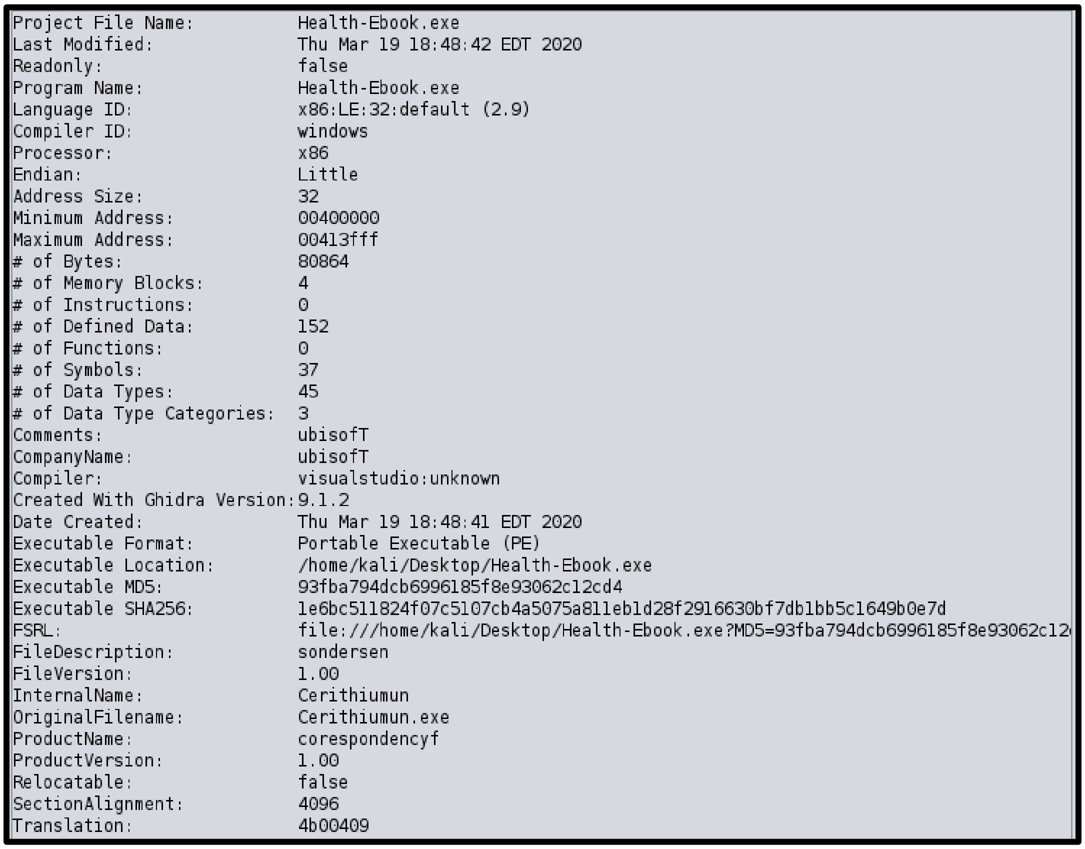
To begin, the malware is loaded into CFF Explorer VIII. The “e_magic” variable is set to the hex value “0x5A4D”, which translates to ASCII, “MZ” (**Figure 10**). This indicates that the file is a **Portable Executable file**, which is used in 32-bit and 64-bit Windows Operating Systems.



e_magic	00000000	Word	5A4D

Figure 10: CFF Explorer e_magic field

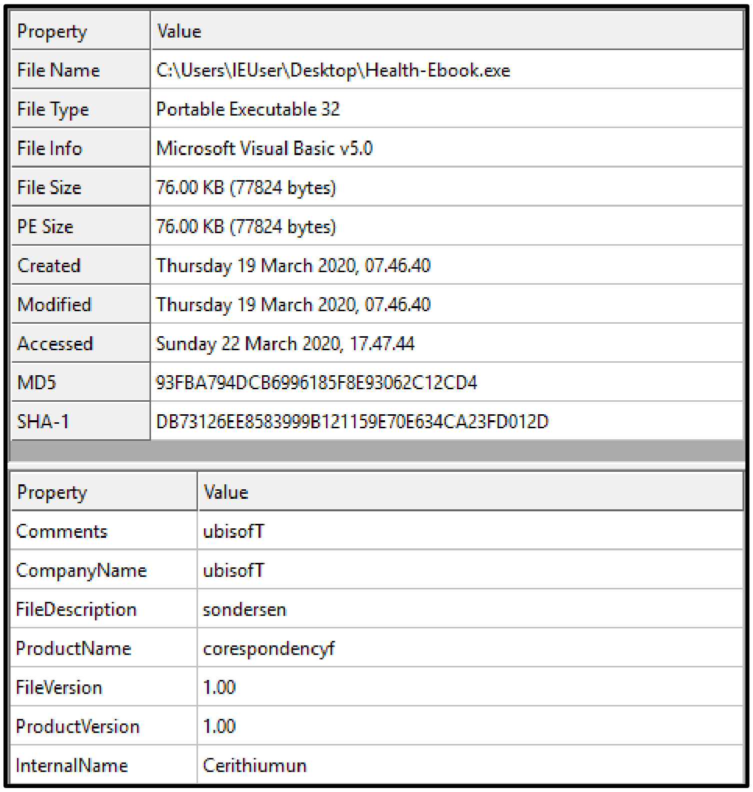
Next, the file is loaded into Ghidra v9.1.2 running on Kali Linux 2020.1a (**Figure 11**). This is done to confirm the previous data that was collected with VirusTotal. The CompanyName field is “**ubisofT**”. The compiler used for this program is **Visual Studio**. The MD5 and SHA1 hashes are confirmed again in Ghidra. The file description is “**sonderson**”. The InternalName is “**Cerithiumun**” and OriginalFilename is “**Cerithiumun.exe**”.



```
Project File Name:           Health-Ebook.exe
Last Modified:               Thu Mar 19 18:48:42 EDT 2020
Readonly:                    false
Program Name:                Health-Ebook.exe
Language ID:                 x86:LE:32:default (2.9)
Compiler ID:                 windows
Processor:                   x86
Endian:                      Little
Address Size:                32
Minimum Address:             00400000
Maximum Address:             00413fff
# of Bytes:                  80864
# of Memory Blocks:          4
# of Instructions:           0
# of Defined Data:           152
# of Functions:              0
# of Symbols:                37
# of Data Types:             45
# of Data Type Categories:   3
Comments:                    ubisofT
CompanyName:                 ubisofT
Compiler:                    visualstudio:unknown
Created With Ghidra Version:9.1.2
Date Created:                Thu Mar 19 18:48:41 EDT 2020
Executable Format:           Portable Executable (PE)
Executable Location:         /home/kali/Desktop/Health-Ebook.exe
Executable MD5:              93fba794dcb6996185f8e93062c12cd4
Executable SHA256:           1e6bc511824f07c5107cb4a5075a811eb1d28f2916630bf7db1bb5c1649b0e7d
FSRL:                        file:///home/kali/Desktop/Health-Ebook.exe?MD5=93fba794dcb6996185f8e93062c12
FileDescription:             sondersen
FileVersion:                 1.00
InternalName:                Cerithiumun
OriginalFilename:            Cerithiumun.exe
ProductName:                 corespondencyf
ProductVersion:              1.00
Relocatable:                 false
SectionAlignment:            4096
Translation:                 4b00409
```

Figure 11: Ghidra Output for Health-Ebook.exe

CFF Explorer lists the compiler as **Microsoft Visual Basic v5.0**. It also confirms the file properties that were analyzed in Ghidra (**Figure 12**).

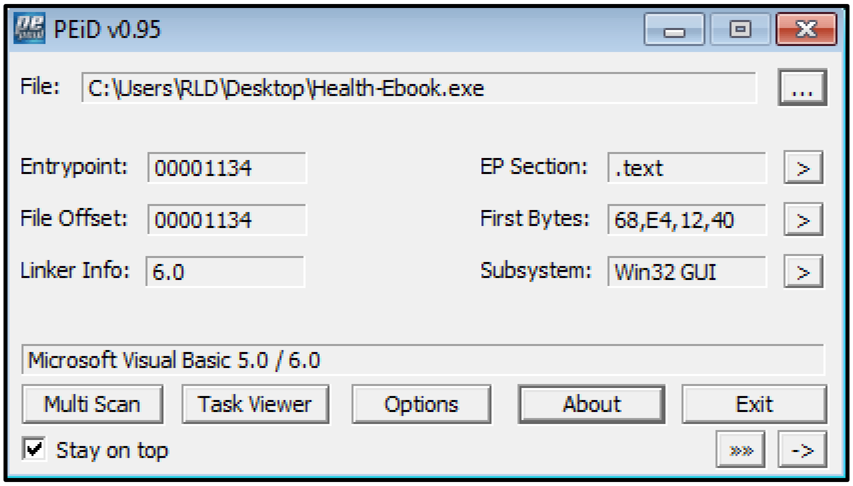


Property	Value
File Name	C:\Users\IEUser\Desktop\Health-Ebook.exe
File Type	Portable Executable 32
File Info	Microsoft Visual Basic v5.0
File Size	76.00 KB (77824 bytes)
PE Size	76.00 KB (77824 bytes)
Created	Thursday 19 March 2020, 07.46.40
Modified	Thursday 19 March 2020, 07.46.40
Accessed	Sunday 22 March 2020, 17.47.44
MD5	93FBA794DCB6996185F8E93062C12CD4
SHA-1	DB73126EE8583999B121159E70E634CA23FD012D

Property	Value
Comments	ubisofT
CompanyName	ubisofT
FileDescription	sondersen
ProductName	corespondencyf
FileVersion	1.00
ProductVersion	1.00
InternalName	Cerithiumun

Figure 12: CFF Explorer Output for Health-Ebook.exe

PEID v 0.95 can be used to further analyze the portable executable file. It confirms the compiler to be **Microsoft Visual Basic 5.0 / 6.0** and the file offset / entry point is at 00001134. The entry point is where the instructions of the program are first executed. This can be seen in **Figure 13**.



PEiD v0.95

File: C:\Users\RLD\Desktop\Health-Ebook.exe

Entrypoint: 00001134

EP Section: .text

File Offset: 00001134

First Bytes: 68,E4,12,40

Linker Info: 6.0

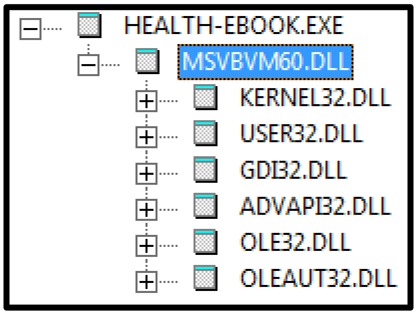
Subsystem: Win32 GUI

Microsoft Visual Basic 5.0 / 6.0

☑ Stay on top

Figure 13: PEID v0.95 Output for Health-Ebook.exe

Using Dependency Walker, the API calls are identified. These files can help provide an understanding on the functionality of the program. To begin, the program has one DLL with 6 other DLLs branching off it. **Figure 14** provides the structure of the program based on the Dependency Walker output.



```
⊟ HEALTH-EBOOK.EXE
    ⊟ MSVBVM60.DLL
        ⊞ KERNEL32.DLL
        ⊞ USER32.DLL
        ⊞ GDI32.DLL
        ⊞ ADVAPI32.DLL
        ⊞ OLE32.DLL
        ⊞ OLEAUT32.DLL
```

Figure 14: Dependency Walker Output for Health-Ebook.exe

Health-Ebook.exe is contained within a process called **MSVBM60**, which is a Visual Basic module. This is a common action that GuLoader malware will take: “[Mapping] the image of a system DLL”, and then “[overwriting] the system DLL image” with its unpacked code (Proofpoint, 2020). By doing this, it can evade antivirus detection. Inside the MSVBVM60.DLL file are other system .DLLs that perform system-related API calls. These calls can be used to conduct many different system related tasks. The following section will describe API calls of interest:

API Calls of Interest

These API calls are responsible for system functions such as accessing credentials or applying privileges to accounts.

AddAce – Adds access control entries to specified Access Control Lists.
AddAccessDeniedAce – This, along with the AddAccessAllowedAce below, refer to Access Control Entries for files: It will either deny access or add access to an Access Control List based on a user's
AddAccessAllowedAce – Listed previously, this adds access for a specific user to access a file using an ACL.
AddUsersToEncryptedFile
AdjustTokenGroups – Enables or Disables Groups related to the current Access Token, which contains information about the current logon session.
AdjustTokenPrivileges – Adjusts user privileges in the current access token.

CredDeleteA – Deletes Credentials from user's credential set
CredReadDomainCredentialsA – A function that reads the domain credentials from the user's domain credential set from the current session: this credential set is of the current token.
CredWriteDomainCredentials – Writes new credentials from the user's domain credential set.
CredUnprotectA – Decrypts credentials that were previously encrypted using CredProtect, which protects the user's logon credentials.
CrewWriteA – Creates or edits users current logon credentials.
CredEnumerate – Enumerates the credentials for the current login.
CredFindBestCredential – Searches the credential database for credentials that match the current logon session.
https://docs.microsoft.com/en-us/windows/win32/secauthn/credentials-management
CredGetTargetInfoA – obtains information about the name of the target computer. Passed to ReadDomainCredentials and WriteDomainCredentials.
CredISProtected – determines if the credentials are encrypted or not (from the CredProtect function)
CredProfileLoaded – Not documented by Microsoft but is related to the CredProtect function. Based on the name, it should be responsible for loading the user profile.
CredProfileUnloaded
CredMarshalCredential
CredProtect – Encrypts the specified credentials (yet another credential encryption API)
CredReadByTokenHandle
CredRestoreCredentials
CryptDeriveKey

LogonUser - Logs a user onto the local computer using their username and password

LookupAccountName - Grabs the Security Identifier for the provided system name

LookupAccountSid - Provides information about a target computer based on its Security Identifier
LookupPrivilegeDisplayName – Looks up display name for privilege requested.
LookupPrivilegeNameA – Looks up the locally unique identifier for the logon session, which has specific privileges.
LookupPrivilegeValue – Referring to privilege changes in the current logon session.
LookupSecurityDescriptorParts – Regarding security information for other functions.
LsaAddAccountRights – Assigns privileges to accounts.
LsaAddPrivilegesToAccount – Not documented by Microsoft.
LsaClearAuditLog – Not documented by Microsoft.
LsaClose
LsaCreateAccount
LsaCreateSecret
LsaCreateTrustedDomain
LsaCreateTrustedDomainEx
LsaDelete
LsaDeleteTrustedDomain
LsaEnumerateAccountRights
LsaEnumerateAccounts
LsaEnumerateAccountsWithUserRight
LsaEnumeratePrivileges
LsaEnumeratePrivilegesOfAccount
LsaEnumerateTrustedDomains – Provides information about the trusted domains of a local system.

Lastly, using the **strings** command in Kali Linux provided some strings of interest:

KULDSLAAEN
Bevaringsforen
Cerithiumun
kejsersnittet
Teatretsboggru5
Agamica
Unastonishgen
genoplivh
Reversibilit3
Eisingaktionr2
fijianeremum
grunter
Tildelend
landlessnesscy
Aabenba
opstign

These strings are in many different languages: most are in Danish, but some are in Norweigian and German. “kejsersnittet” in Danish translates to “Caesarean”, possibly referring to the caesarean cipher. “Eisingaktionr2” is German for “Ice Action 2”. “Reversibilit” is Danish for “Reversibility”. “KULDSLAAEN” is simply a Norwegian name. Although these strings were found in the .exe file, they aren’t referenced anywhere else.

Dynamic Analysis

To begin, Any.Run and JoeSandbox are used to automate the dynamic analysis process. These cloud-based sandboxes will run the malware in a safe environment and analyze its functions. The will run the malware in the following environments: Windows 7 Professional Service Pack 1 (build: 7601, 32 bit) (Any.Run) and Windows 10 64 bit (version 1803) with **Office 2016**, Adobe Reader DC 19, Chrome 70, Firefox 63, Java 8.171, Flash 30.0.0.113 (Joe Sandbox). Also, URL Revealer by Kahu Security and FiddlerCap Web Recorder are used in both Windows 7 and Windows 10 virtual machines.

Initial Execution: Flow of Events

Health-Ebook.exe and Process Hollowing

The initial Health-Ebook.exe contains a PID of 3764. This process uses the **HideFromDebugger** thread set, providing anti-debugging capabilities. Upon execution, Health-Ebook.exe spawns a process named after itself: Health-Ebook.exe. This second process contains a PID of 1616. This technique is called **process hollowing**. This happens when “a process is created in a suspended state then its memory is unmapped and replaced with malicious code” (MITRE, 2017). This is a type of defense evasion technique that the malware is using to remain undetected. Process 1616 is responsible for contacting a Google service, googhlehosted.I.googleusercontent.com over HTTPS. This can be seen in **Figure 15**.

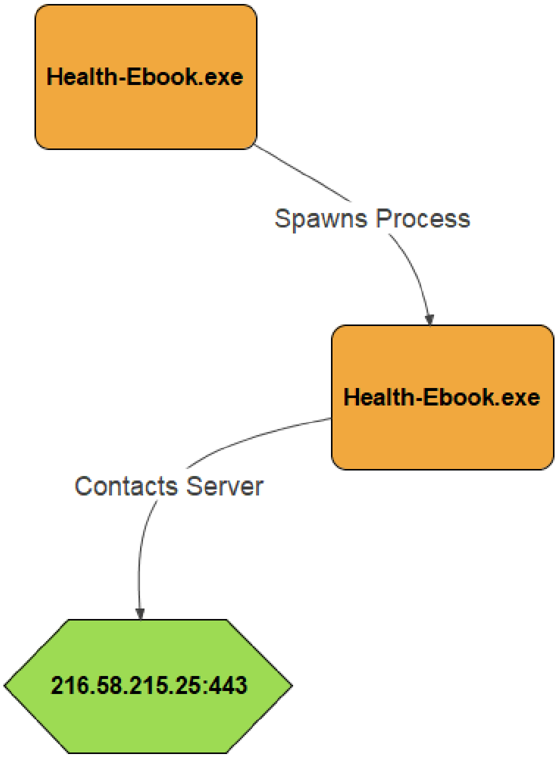


Figure 15: Health-Ebook.exe Process Hollowing

Next, the newly spawned Health-Ebook.exe spawns explorer.exe, which spawns more processes (**Figure 16**).

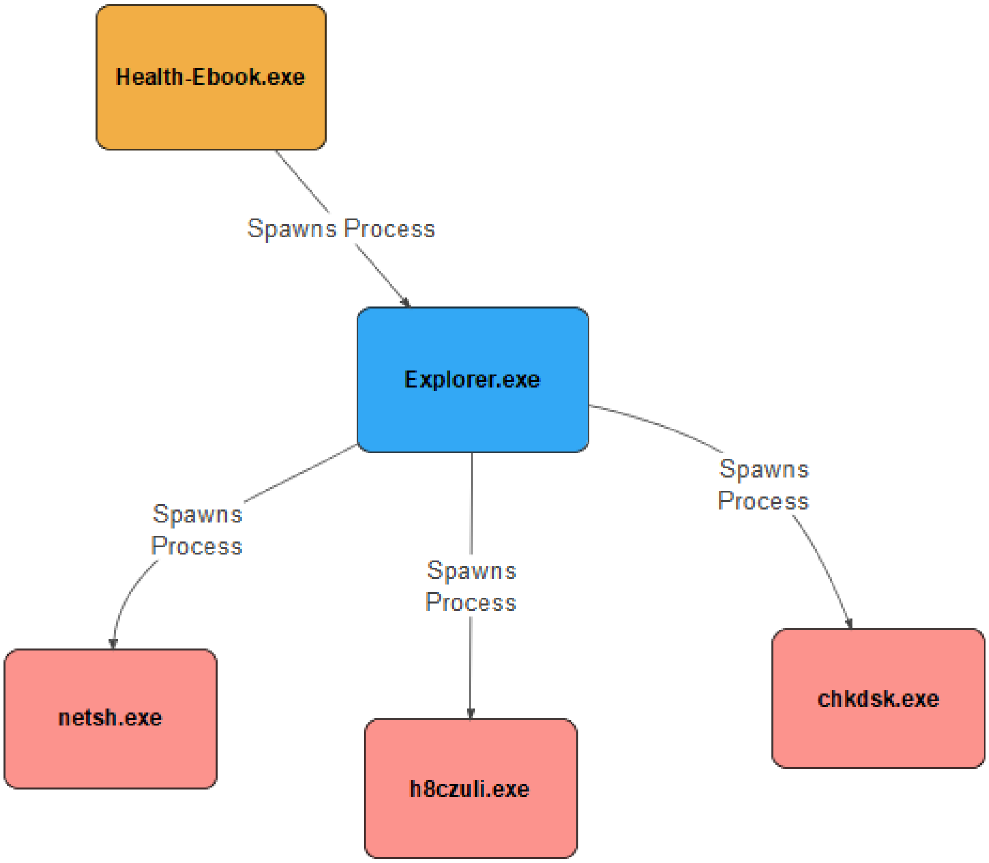


Figure 16: Health-Ebook.exe spawning explorer.exe

explorer.exe Process

explorer.exe spawns **netsh.exe**, **h8czuli.exe**, and **chkdsk.exe**. It also contacts two domains: **www.kbasherphotography[.]com** (192.0.78.24 over Port 49749, 49750, and 80) and **www.michalshahar[.]com** (162.209.159.116 over Port 49745 and 80). These two domains are potential Command and Control (C2) servers, which allow the attacker to remotely access and send commands to the infected machine.

netsh.exe Process

netsh.exe is a process that is often used by Formbook malware, a type of malware that is known to steal login credentials. In this example, this process does just that: It opens registry keys containing form data for Mozilla Firefox, Mozilla Thunderbird, Outlook, and more. It also opens a registry key in HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\IntelliForms\Storage2, which is responsible for storing auto-complete passwords in the browser. Lastly, it copies C:\Users\user\AppData\Local\Google\Chrome\User Data\Default\Login Data. In addition to this Formbook activity, Netsh.exe also creates a registry key to establish persistence at startup, **h8tczuli.exe**, in the location HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run. This is a persistence tactic used to ensure that the process survives a reboot: this will cause the program referenced to be executed when a user logs in" (MITRE, 2017). It also creates two cmd.exe processes, which can be seen in **Figure 17**.

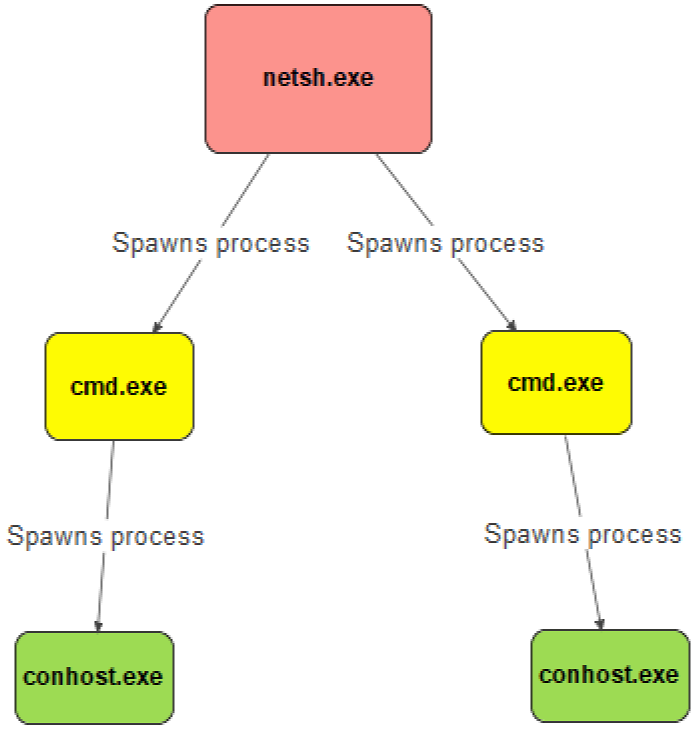


Figure 17: netsh.exe spawning two cmd.exe processes that spawn conhost.exe processes

The first **cmd.exe** process (PID 1108) is used to collect Google Chrome user data from the User Data\Default\Login Data folder. This is copied into a temporary folder in the user's

Local\Temp\DB1 folder. This process spawns a **conhost.exe** process (PID 1458). **conhost.exe** is "used to transfer messages between console clients and servers" (Davis 2017). The attacker could potentially be using this process to communicate with the infected machine remotely.

The second **cmd.exe** process (PID 2084) tries to delete Health-Ebook.exe from the directory that it was executed in. There is a flaw in the code that prevents it from deleting the original file. It also spawns a **conhost.exe** process (PID 1456) that serves the same function as the previous conhost.exe.

h8tczuli.exe Parent and Child Process

The next function is **h8tczuli.exe** (PID 3224). This process has the "HideFromDebugger" flag set. This is a procedure that is used to prevent debugging during dynamic analysis. Also, the process reads data from C:\Users\user\AppData\Local\Microsoft\Windows\INetCookies and \INetCache, which store the user's cache and cookies for Internet Explorer. Lastly, is process is set to run at startup, so it survives a reboot.

This process spawns a process of itself (process hollowing, again) named **h8tczuli.exe** (**Figure 18)**

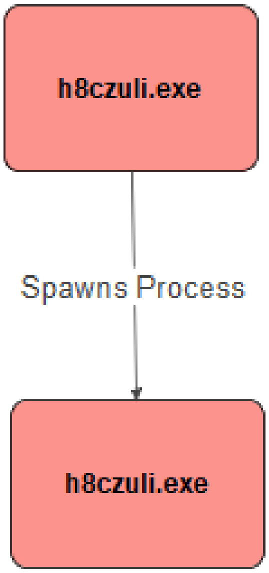


Figure 18: h8tczuli.exe Process Hollowing

The child process **h8tczuli.exe** (PID 612) reads the same Internet Explorer cache and cookie files that its parent process reads. It also appears to contact two Google domains (**Figure 19**).

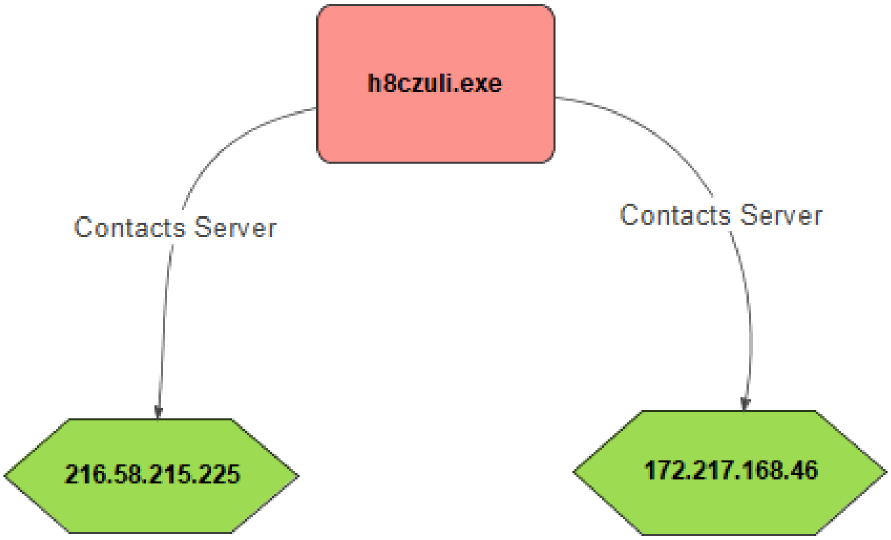


Figure 19: h8czuli.exe child process contacting 2 Google domains

The last process spawned by the malware is **chkdsk.exe** (PID 484). This process is stored in C:\Windows\SysWOW64\chkdsk.exe. The sole purpose of this function is to intercept Read Time Stamp Counter (RDTSC) instructions, which are responsible for CPU timing in a virtualized environment. By understanding the CPU timing, the malware can determine if it is being executed in sandbox or not.

Figure 20: chkdsk.exe, responsible for sandbox evasion

Network Activity

This malware contacts four active web pages. The websites visited are www.michalshahar[.]com (**162.209.159.116**), www.aeaco[.]net (**63.250.33.106**) and www.kbasherphotography[.]com (**192.0.78.24**). In addition to this, it also contacts the site googlehosted.l.googleusercontent.com at (**216.58.215.225**). Using Maltiverse Threat Analyzer tool, the URLs were be analyzed for malicious indicators. JoeSandbox also provides some insight on these IP Addresses.

User Agent

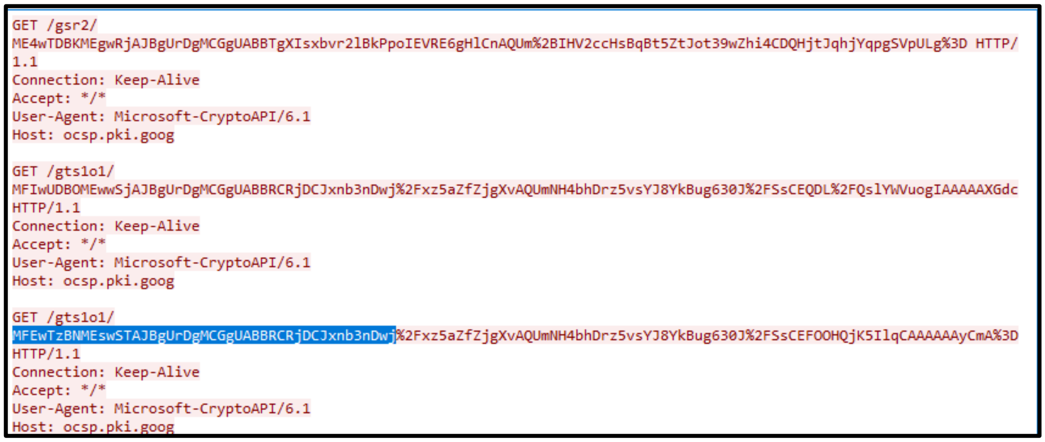
Next, the malware uses a web browser user-agent that is commonly used with other forms of malware: User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; Trident/7.0; rv:11.0).

IP Address / URL Analysis

To begin, **192.0.78.24** has been blacklisted by Hybrid-Analysis, OpenPhish, Maltiverse Research Team, and others for the following actions: serving trojans, containing CVE exploits, phishing for Apple Credentials, phishing for cryptocurrency wallet addresses, serving adware, and serving ransomware. The Threat Analysis tool even lists of specific types of malicious software served: Locky Ransomware, VBObfus-G Trojan, Fuery-C Trojan, AD.Swotter, Razy, and more. **63.250.33.106** was blacklisted for serving malware, specifically Kryptic.BEV.gen. Lastly, the Google Hosted IP Address **216.58.215.225** was interesting. According to the JoeSandbox analysis, this IP Address previously served a COVID-19 themed file: the file is named "COVID -19 Treatment & Cure.pptx". At the time of this writing, no cure exists for COVID-19.

Wireshark .pcap Analysis

Analyzing the Wireshark packet capture provided by the Any.Run analysis, it can be noticed that the malware makes three HTTP GET Requests. The user agent for these requests are Microsoft CryptoAPI / 6.1 and the resources that are being retrieved are unintelligible (**Figure 21**).



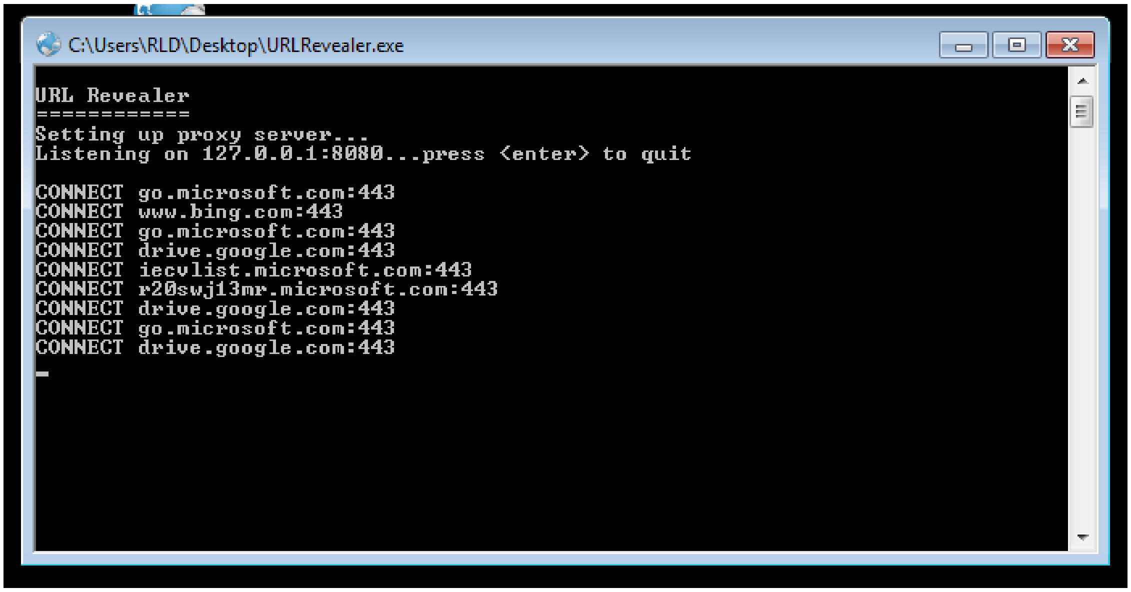
```
GET /gsr2/
ME4wTDBKMEgwRjAJBgUrDgMCGgUABBTgXIsxbvr2lBkPpoIEVRE6gHlCnAQUm%2BIHV2ccHsBqBt5ZtJot39wZhi4CDQHjtJqhjYqpgSVpULg%3D HTTP/
1.1
Connection: Keep-Alive
Accept: */*
User-Agent: Microsoft-CryptoAPI/6.1
Host: ocsp.pki.goog

GET /gts1o1/
MFIwUDBOMEwwSjAJBgUrDgMCGgUABBRCRjDCJxnb3nDwj%2Fxz5aZfZjgXvAQUmNH4bhDrz5vsYJ8YkBug630J%2FSsCEQDL%2FQslYWVuogIAAAAAXGdc
HTTP/1.1
Connection: Keep-Alive
Accept: */*
User-Agent: Microsoft-CryptoAPI/6.1
Host: ocsp.pki.goog

GET /gts1o1/
MFEwTzBNMEswSTAJBgUrDgMCGgUABBRCRjDCJxnb3nDwj%2Fxz5aZfZjgXvAQUmNH4bhDrz5vsYJ8YkBug630J%2FSsCEFOOHQjK5IlqCAAAAAAyCmA%3D
HTTP/1.1
Connection: Keep-Alive
Accept: */*
User-Agent: Microsoft-CryptoAPI/6.1
Host: ocsp.pki.goog
```

Figure 21: Wireshark .pcap analysis of network traffic inside Windows 8 Virtual Machine

Unfortunately, the attacker uses an obfuscation technique on the GET requests seen in the image above, making the requests incomprehensible.

URL Revealer Tool

Using the URL Revealer tool by Kahu Security, the actual links can be seen. This tool allows the request to be made but drops the connection before it can successfully communicate or download files. This tool was used in both Windows 7 and Windows 10 virtual machines. The results can be seen in **Figure 22**:



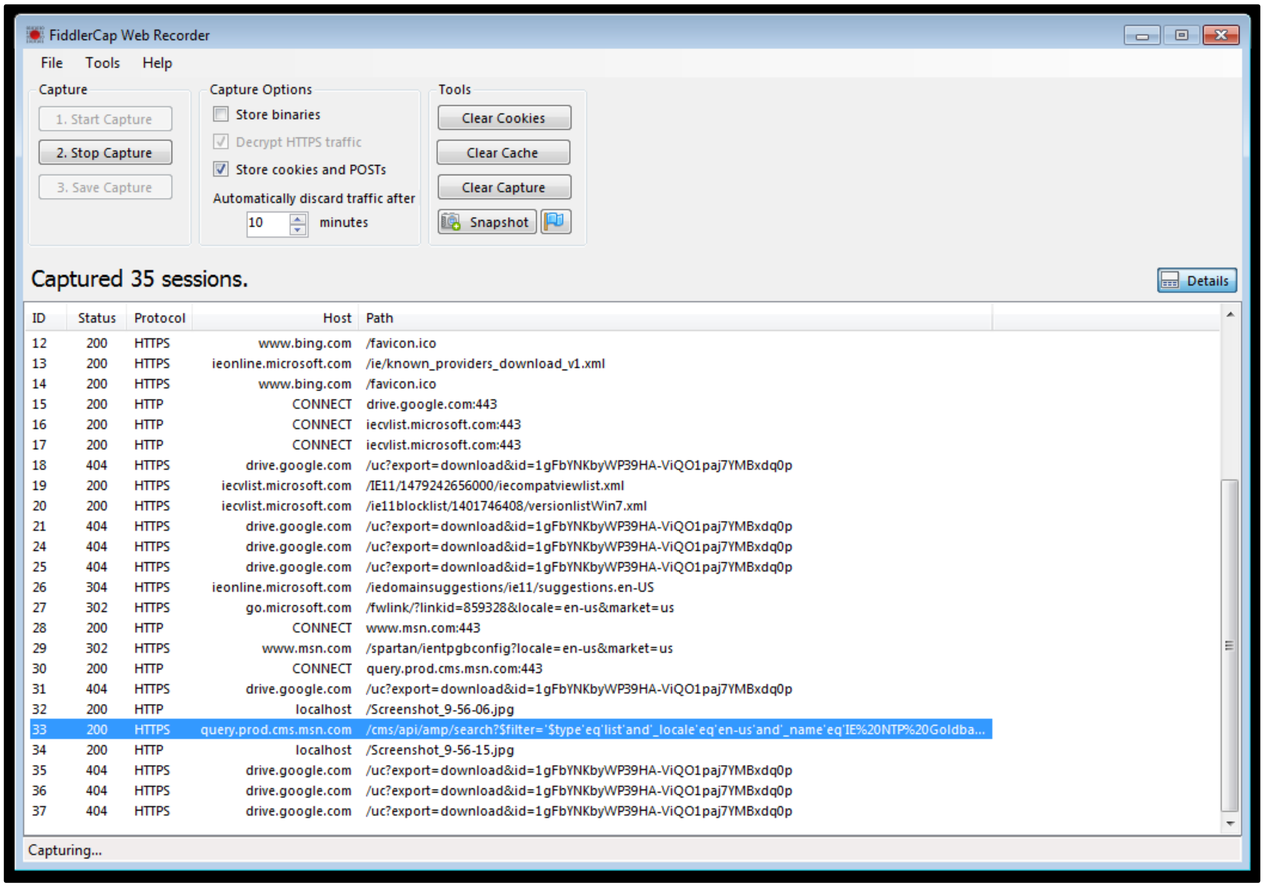
```
URL Revealer
============
Setting up proxy server...
Listening on 127.0.0.1:8080...press <enter> to quit

CONNECT go.microsoft.com:443
CONNECT www.bing.com:443
CONNECT go.microsoft.com:443
CONNECT drive.google.com:443
CONNECT iecvlist.microsoft.com:443
CONNECT r20swj13mr.microsoft.com:443
CONNECT drive.google.com:443
CONNECT go.microsoft.com:443
CONNECT drive.google.com:443
_
```

Figure 22: URLRevealer Output in Windows 8 Machine

Both versions of Windows had the same results: connections being made to drive.google.com. Google Drive has been recently abused to evade detection and download encrypted payloads.

FiddlerCap Web Recorder

To confirm these visited URLs, FiddlerCap Web Recorder is used in both Windows 8 and Windows 10. This software also captures requests and provides the corresponding URL (**Figure 23**).



☐ Store binaries
☑ Decrypt HTTPS traffic
☑ Store cookies and POSTs
Automatically discard traffic after 10 minutes

Captured 35 sessions.

ID	Status	Protocol	Host	Path
12	200	HTTPS	www.bing.com	/favicon.ico
13	200	HTTPS	ieonline.microsoft.com	/ie/known_providers_download_v1.xml
14	200	HTTPS	www.bing.com	/favicon.ico
15	200	HTTP	CONNECT	drive.google.com:443
16	200	HTTP	CONNECT	iecvlist.microsoft.com:443
17	200	HTTP	CONNECT	iecvlist.microsoft.com:443
18	404	HTTPS	drive.google.com	/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdq0p
19	200	HTTPS	iecvlist.microsoft.com	/IE11/1479242656000/iecompatviewlist.xml
20	200	HTTPS	iecvlist.microsoft.com	/ie11blocklist/1401746408/versionlistWin7.xml
21	404	HTTPS	drive.google.com	/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdq0p
24	404	HTTPS	drive.google.com	/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdq0p
25	404	HTTPS	drive.google.com	/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdq0p
26	304	HTTPS	ieonline.microsoft.com	/iedomainsuggestions/ie11/suggestions.en-US
27	302	HTTPS	go.microsoft.com	/fwlink/?linkid=859328&locale=en-us&market=us
28	200	HTTP	CONNECT	www.msn.com:443
29	302	HTTPS	www.msn.com	/spartan/ientpgbconfig?locale=en-us&market=us
30	200	HTTP	CONNECT	query.prod.cms.msn.com:443
31	404	HTTPS	drive.google.com	/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdq0p
32	200	HTTP	localhost	/Screenshot_9-56-06.jpg
33	200	HTTPS	query.prod.cms.msn.com	/cms/api/amp/search?$filter='$type'eq'list'and'_locale'eq'en-us'and'_name'eq'IE%20NTP%20Goldba...
34	200	HTTP	localhost	/Screenshot_9-56-15.jpg
35	404	HTTPS	drive.google.com	/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdq0p
36	404	HTTPS	drive.google.com	/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdq0p
37	404	HTTPS	drive.google.com	/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdq0p

Capturing...

Figure 23: FiddlerCap Web Recorder Output on Windows 8 Machine

It can be clearly seen that this malware is attempting to download multiple files via Google Drive, but is unable to do so.

Malware Summary

Observing the overall behavior of this file, it resembles VB5/6 GuLoader. This type of malware was discovered in December 2019 by Proofpoint researchers. The first few steps after execution are evident of this. As previously stated, the malware "spawns a child process copy of itself", "maps the image of a system DLL" (msvbm60.dll), and "injects the unpacking code into the child" (Proofpoint Threat Research Team). This process makes GuLoader very difficult to analyze, but dynamic analysis can be used to understand its behavior. The malware reaches out to Google drive to download an encrypted payload, which is a common characteristic of

GuLoader. In this case, the GuLoader downloads Formbook malware, which steals form data from web browsers installed on the system (Google Chrome, Internet Explorer, Firefox).

IOCS

192.0.78.24
63.250.33.106
93fba794dcb6996185f8e93062c12cd4
db73126ee8583999b121159e70e634ca23fd012d
1e6bc511824f07c5107cb4a5075a811eb1d28f2916630bf7db1bb5c1649b0e7d
drive.google[.]com/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdp0p
http://www.michalshahar[.]com/w0k/?r65hj=BN90bfcptvP4SJ&3fct=vO9Vm2RARflm5p1PFXqn6eBrWTFFnunBf6X3DMkFEdmGbjkCk/pABuPtOpuxvLvCis20
http://www.kbasherphotography[.]com/w0k/?r65hj=BN90bfcptvP4SJ&3fct=3PkPLEV8daGFL4/3pxhg1tKv6aVypEBkpsp65f+Yzy4XBcektFNWUD7dAcSGsTOSbbgw

YARA Rule

```
Rule MyHealth_GuLoader
{
meta:
        description = “Rule to detect the MyHealth GuLoader on Windows machines”

strings:
        $a = “drive.google.com/uc?export=download&id=1gFbYNKbyWP39HA-
ViQO1paj7YMBxdp0p”
        $b =
“http://www.michalshahar.com/w0k/?r65hj=BN90bfcptvP4SJ&3fct=vO9Vm2RARflm5p1PFXq
n6eBrWTFFnunBf6X3DMkFEdmGbjkCk/pABuPtOpuxvLvCis20”
        $c = “http://www.kbasherphotography.com/w0k/”
        $d =
http://www.kbasherphotography.com/w0k/?r65hj=BN90bfcptvP4SJ&3fct=3PkPLEV8daGFL4/3
pxhg1tKv6aVypEBkpsp65f+Yzy4XBcektFNWUD7dAcSGsTOSbbgw
        $mz = {4d 5A}

condition: ($a or $b or $c or $d and $mz)
}
```

References

Adams, L. (2020) Data-Stealing FormBook Malware Preys on Coronavirus Fears. Retrieved from https://www.bleepingcomputer.com/news/security/data-stealing-formbook-malware-preys-on-coronavirus-fears/

Carnegie Mellon University Information Security Office (2020). Phishing: Don't Be the Latest Catch. Retrieved from https://www.cmu.edu/iso/news/2020-news/phishing-dont-be-the-latest-catch.html

Davis, A (2020). Monitoring Windows Console Activity (Part 1). Retrieved from https://www.fireeye.com/blog/threat-research/2017/08/monitoring-windows-console-activity-part-one.html

Dudley, T (2018) OUCH! Newsletter: Stop That Phish. Retrieved from https://www.sans.org/security-awareness-training/resources/stop-phish

Elledge, A. (2004) Phishing: An Analysis of a Growing Problem, page 9. Retrieved from https://www.sans.org/reading-room/whitepapers/threats/paper/1417

Krebs, B. (2020) Live Coronavirus Map Used to Spread Malware. Retrieved from https://krebsonsecurity.com/2020/03/live-coronavirus-map-used-to-spread-malware/

MITRE Corporation (2017). Process Hollowing.
Retrieved from https://attack.mitre.org/techniques/T1093/

MITRE Corporation (2017) Registry Run Keys / Startup Folder
Retrieved from https://attack.mitre.org/techniques/T1060/

Muncaster, P (2020) Trickbot Named Most Prolific #COVID19 Malware. Retrieved from https://www.infosecurity-magazine.com/news/trickbot-named-most-prolific/

Proofpoint Threat Research Team (2020) GuLoader: A Popular New VB6 Downloader that Abuses Cloud Services. Retrieved from https://www.proofpoint.com/us/threat-insight/post/guloader-popular-new-vb6-downloader-abuses-cloud-services

Saleh, T. (2020) CovidLock: Mobile Coronavirus Tracking App Coughs Up Ransomware. Retrieved from https://www.domaintools.com/resources/blog/covidlock-mobile-coronavirus-tracking-app-coughs-up-ransomware#