THEMED PHISHING

A Deep Dive into a Multifaceted COVID-19 Phishing Campaign

Robert Dean

Marshall University: Cyber Forensics and Security

Table of Contents

Introduction

During the event of any major tragedy, hackers have exploited human emotion and used themed phishing campaigns to profit from innocent people. Phishing attacks date all the way back to the 1990s when hacker groups attempted to steal AOL "member's password and account information … by posing as" employees of the service provider organization (Stutz, 1997). Shortly after 9/11, attackers used the World Trade Center tragedy that took place in America to their advantage, leveraging phishing attacks against US citizens to conduct fake 'ID Checks' (Miller, 2008). December 2019 marks the tragedy known as COVID-19, a virus that has infected millions of people across the world. Nearly every country has been infected with the virus, making all of them targets for themed phishing attacks. The attackers took advantage of the crisis that was unraveling: many hackers utilized email, fake websites, and malware to steal information from innocent people. They have impersonated the World Health Organization along with many government organizations to do so. They have utilized various forms of malware to infect computers across the world, stealing login data and other personal data.

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 a trusted party or organization (the World Health Organization, for example). They can simply trick the user into sending over login information or personally identifiable information (PII) by using deceiving websites or emails. This phishing tactic is referred to as social engineering, where the "hoax e-mails used in phishing schemes allege to be from a trusted entity" (Elledge, 2007, p. 9). They gain the user's trust and exploit that weakness to obtain information. Phishing

attacks are often used to deliver malware. The malware, which may look like a real program, can be performing malicious processes in the background, such as stealing credit card information, account passwords, and more.

During the COVID-19 epidemic, hackers have used may different platforms to steal personal data. In one case, they used a mobile application available on Android that appears to be a map of COVID-19 cases across the U.S. This piece of software is ransomware that encrypts the phone's data, "[requesting] $100 in bitcoin in 48 hours on the ransom note" (Saleh, 2020). This same type of COVID-19 map application was implemented on Windows machines. The Windows version "has [a] real working interactive Coronavirus data map and a payload" that can infect targeted machines with malware (Krebs, 2020). Hackers are exploiting human fear and interest to urge them into downloading the malicious software. In another example, attackers have tricked users into installing Trickbot malware, which is used "to drop additional malware like ransomware, VNC clients and remote access malware" (Muncaster 2020). This type of malware is known as a 'dropper'. Their techniques become more advanced as they progress in a phishing campaign. They attempt the 'easier' attacks first, and if they don't succeed, they try more manipulative and advanced methods of attack.

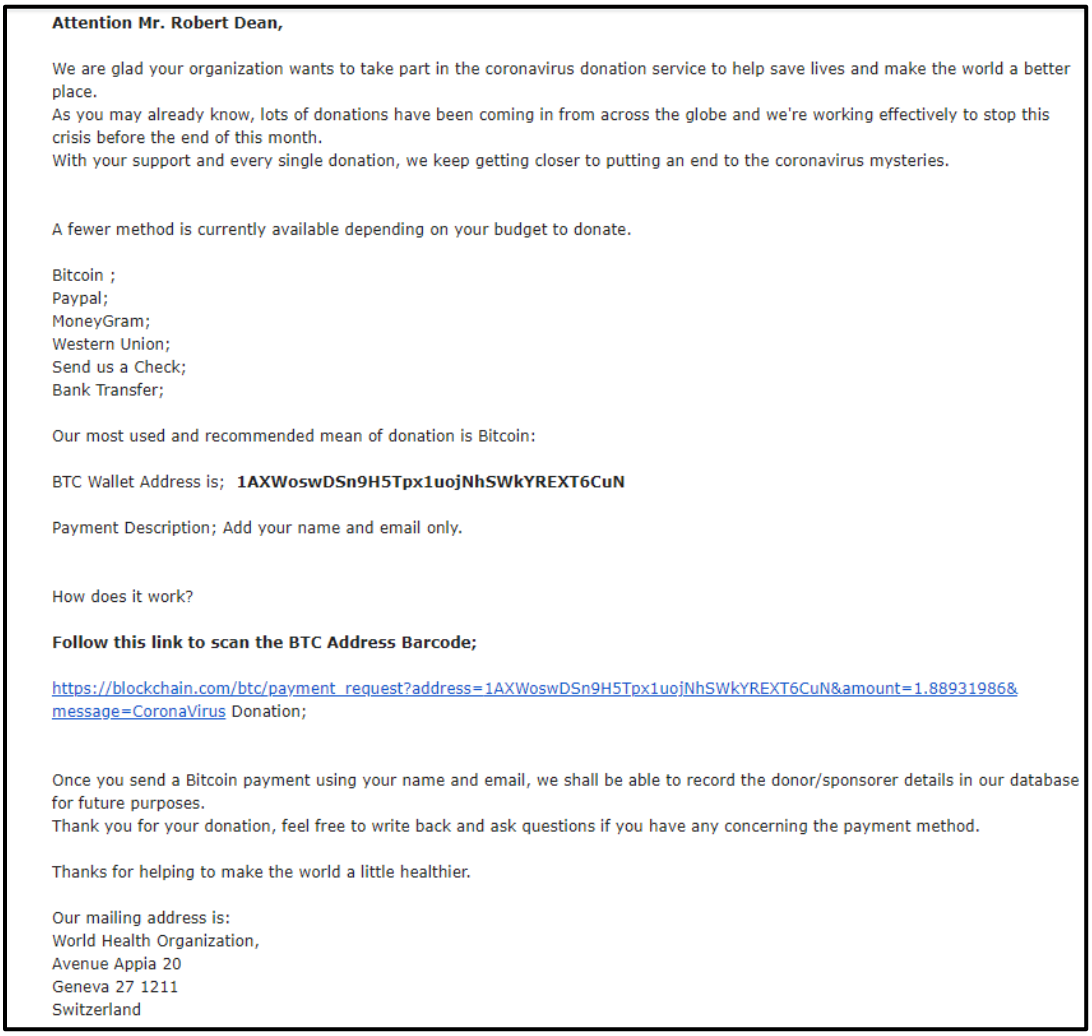
The purpose of this paper is to dive into a COVID-19 themed phishing attack that was used against the U.S. population. This attack combines emails, a fake website, and malware that's sole purpose is to steal user data. This attack took place on Tuesday, March 17th at 5:18 AM. The malware utilized in the phishing attack is called GuLoader, which was newly discovered in March of 2020. This malware utilizes cloud shares to download encrypted payloads, which will then be run to conduct various malicious functions on the local machine. This specific GuLoader downloads FormBook malware, which steals user login information

from web-browser forms. In addition to this, the emails and their contents will be analyzed. This will provide an understanding of how attackers utilize many areas of technology to steal information.

Attack Storyline

As previously stated, this attack begins with a series of emails that will be used to attract the victim. Like many phishing attacks, they attempt to play on human emotion and provide a sense of urgency so that the victim will act quickly. In this example, they try to get the victim to download a piece of malware and get them to visit their phishing website, which imitates a real government website. If one form of attack doesn't work, they try a different, more complex attack. This can be seen in the following series of emails. See **Figure 1**, **4**, and **5**. These figures are screenshots of the emails and do not contain hyperlinks.

Phishing Email One:



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. The attacker claims that if the recipient sends over a donation, it will help save lives and help end the COVID-19 crisis. They even provide the legitimate mailing address to the World Health Organization in the end of the email. It's a simple scam, requesting the user to send them money using Bitcoin, Paypal, MoneyGram, Western Union, Check, or Bank Transfer. 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 cannot 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 in the email displays the initial request made by the attacker, including the specific amount that they requested. This bitcoin request is hosted on blockchain.com/btc, which is a website where bitcoin transfers can be made (see **Figure 2** below).

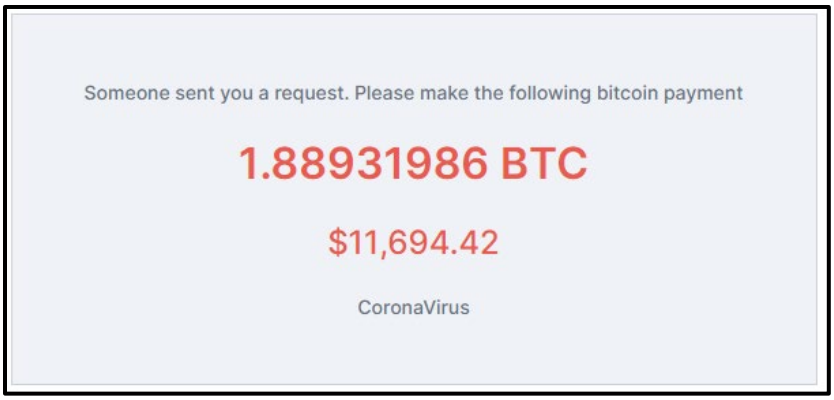


Figure 2: blockchain.com link provided by the attacker used to receive anonymous bitcoin transfers

Lastly, the attacker included the information "WHO: Donations & Grant Office" in the 'From' line in this email, pretending to belong to the World Health Organization (see **Figure 3** below).

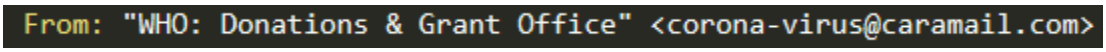


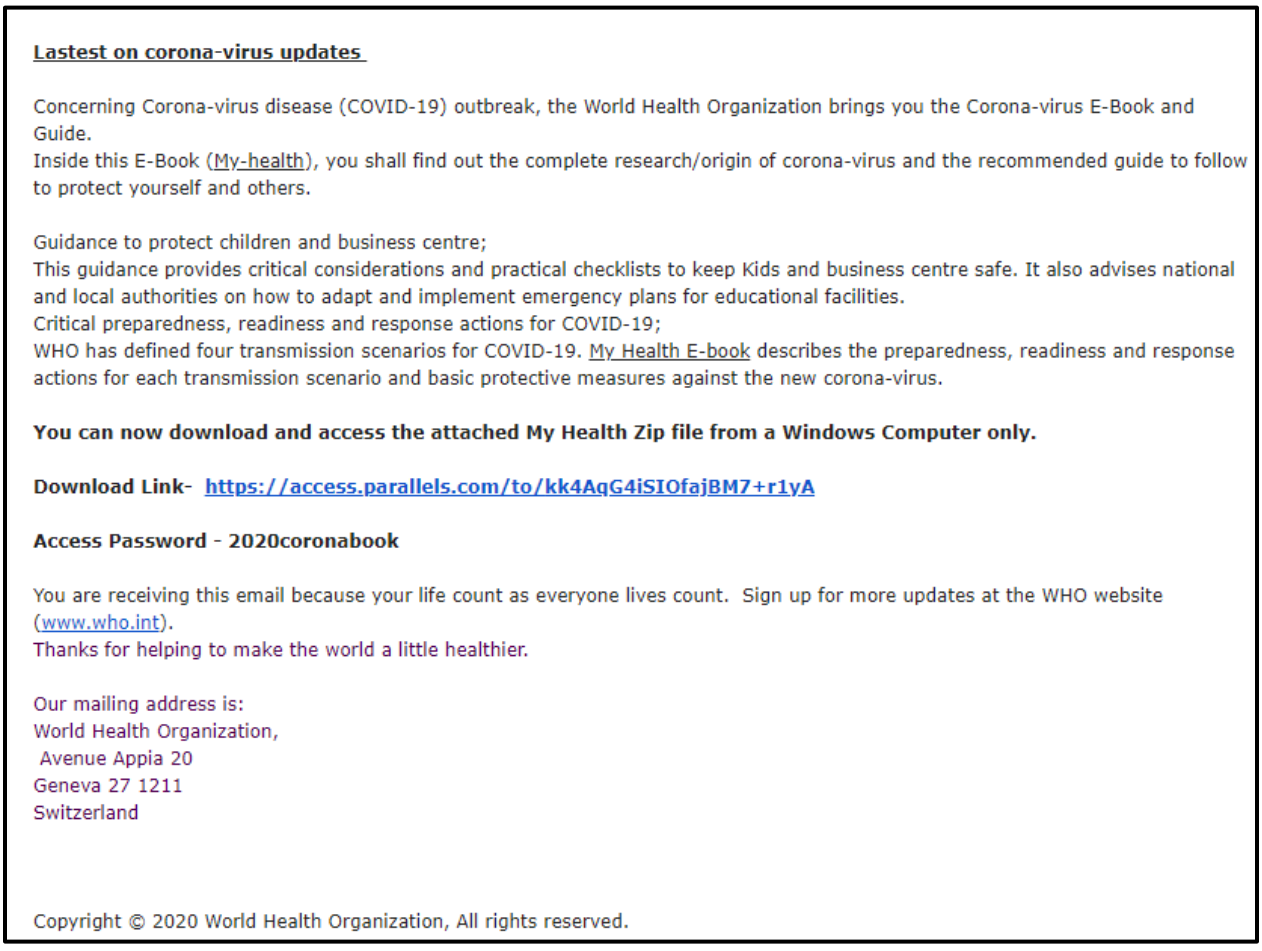
Figure 3: 'From' line included in email header of Phishing Email 1

Also, the email corona-virus@caramail.com is used. This email address does not belong to the World Health Organization, as they use commonly use the email domain "@who.int". Searching the web for the email "corona-virus@caramail.com" reveals its use in a phishing campaign that began in early March of 2020. Apparently, the attackers had posed as the World Health Organization, attempting to distribute an "executable called MyHealth.exe" (Abrams, 2020). This is the first stage of the attack. Most victims will not fall for this scam, because this is obviously illegitimate. It has spelling errors, comes from a suspicious email address, and requests

money via wire-transfer or other means that would not normally be used by the federal government.

Phishing Email Two:

The second part of this phishing campaign involves the delivery of a malicious executable. The attacker again, pretends to be a member of the World Health Organization, providing the WHO mailing address at the end of the email. The attacker introduces the 'new' Corona-virus E-Book and Guide, named "My-Health". They claim the document provides "critical considerations and practical checklists to keep Kids and business centre safe", attempting to trigger a response based on human emotion (see **Figure 4** below).



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

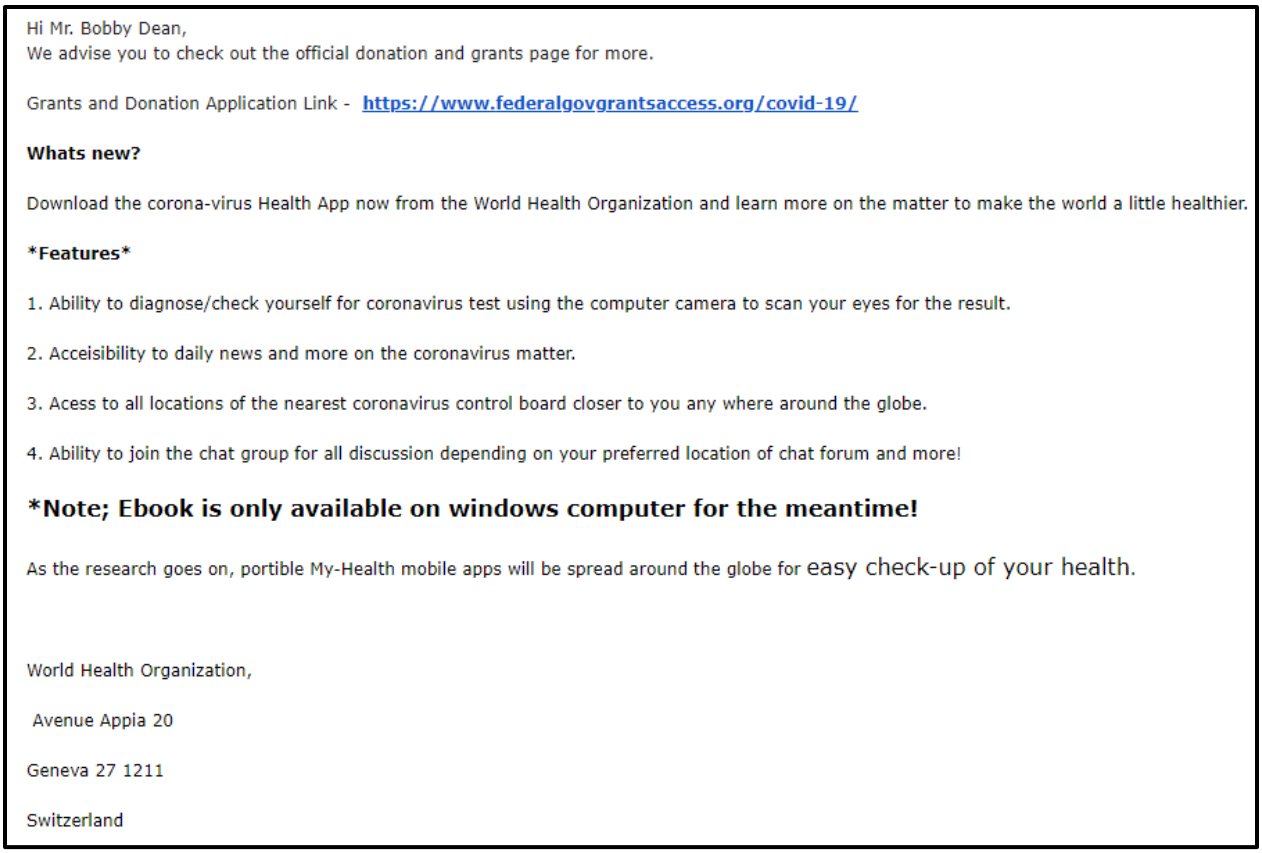
Copyright © 2020 World Health Organization, All rights reserved.

Figure 4: Second Email in Phishing Campaign that includes a link to download Health-Ebook.exe

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 is password protected with the password “2020coronabook”. The file is actually named Health-Ebook.zip. This file will be analyzed in the **malware analysis** section. Indicators that this email is malicious are the spelling and formatting errors, as well as the link to the file (non-government website and not a WHO website).

Phishing Email Three:

Next, a third email was set by the attacker that included another link: https://www.federalgovgrantsaccess.org/covid-19/. The attacker suggested that the victim visits the link to learn more about donations and grants. Next, the attacker encouraged the victim to download the Corona-Virus Health App from the World Health Organization (referring to the previous email). The attackers discussed the ‘features’ of this application, which appear to be very unrealistic in nature.



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 suspicious. One of the 'features' of the application is not even possible: scanning a person's eyes to provide a diagnosis for COVID-19. Another interesting thing to note is the attacker states that the "Ebook is only available on windows computer for the meantime", trying to provide a sense of urgency. Many attackers will "convey a sense of urgency in order to further increase the targets' odds of taking the bait" (Carnegie Mellon 2020). In addition to this, they also state that the software is available for Windows only (because the malware they are pushing only runs on computers running the Windows operating system). Lastly, they prompt the user to go to a suspicious grant website: an analysis of the website provided will be included in the **Phishing Website** section.

Phishing Website

Upon visiting the website, it seems 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/ grant website

Its homepage includes information on COVID-19, the Grand and Cooperative Agreement Act, and provides the stages for which grant policies are made. The contact email at the bottom of the page is covid-19@federalgovgrantaccess.org and the phone number field is left blank (**Figure 7**). A legitimate government website would include their contact phone number, so this is an indicator of a fake website.

Figure 7: Contact Information for Federalgovgrantsaccess.org

The bottom of the homepage displays the text "Packet Pass © 2020". This appears to be a potentially fake organization. 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**).

Figure 8: Full Database Download Link

Unfortunately, the "Full Database" page presents a 404 Not Found page, which indicates that the page does not actually exist or isn't active. On the 'Contact' page, they provide the address for the World Health Organization and a map: this map does not show the correct location. It shows an overview of Brooklyn, New York.

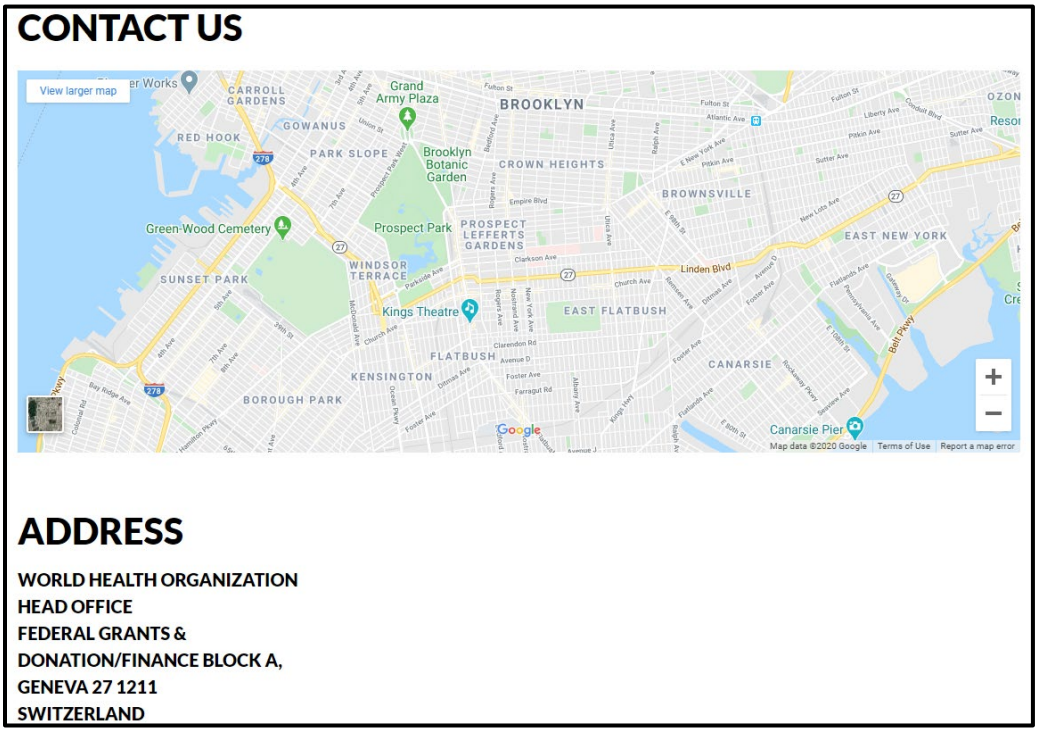


Figure 9: Contact Page containing map of Brooklyn, NY

The flaws associated with this webpage give the appearance that it is a phishing site. Most of the links are broken and there is no evidence proving 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.

Malware Analysis

The name of the malware 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. It is also analyzed using JoeSandbox and Any.Run.

The file was first uploaded on VirusTotal.com, providing the following information:

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**.

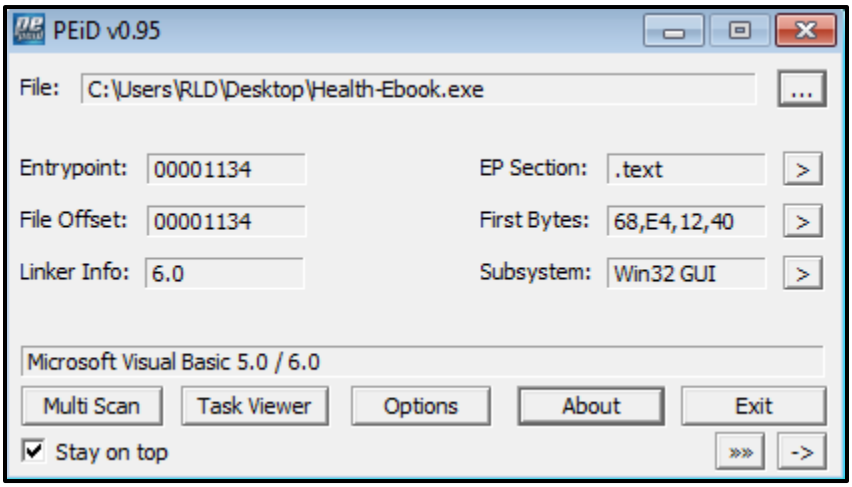


Figure 13: PEID v0.95 Output for Health-Ebook.exe

Using Dependency Walker, the API calls are identified. These files can help provide an understanding on the functionality of the program. This program also provides the overall structure 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.

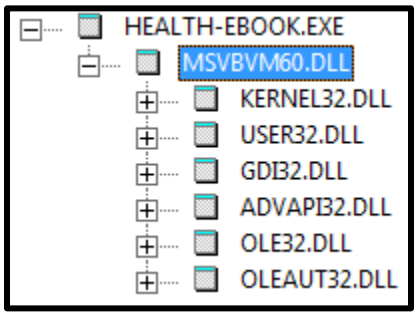


Figure 14: Dependency Walker Output for Health-Ebook.exe

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, defined in the next section.

API Calls of Interest

API calls are Application Programming Interface calls, which allow applications on a computer to interact with each other. These API calls are responsible for system functions such as accessing credentials or applying privileges to accounts. Most API calls are documented on Microsoft's website at the following link: https://docs.microsoft.com/en-us/windows/win32/api/. Many of these API calls refer to privileges of a user, creating accounts on the local domain, enumerating information about accounts on the domain, and more. These API calls can be

utilized by the attacker to escalate privileges. Below are the API calls of interest and a short description of what they do:

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 ID.

AddAccessAllowedAce – Listed previously, this adds access for a specific user to access a file using an ACL.

AddUsersToEncryptedFile – Adds user keys to an encrypted file named pEncryptionCertificates.

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.

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.

CredProtect – Encrypts the specified credentials (yet another credential encryption API).

CryptDeriveKey – Creates a cryptographic key for the current session.

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.

LsaCreateAccount – Creates an account on the domain.

LsaCreateTrustedDomain – Creates a trusted domain on the system.

LsaDeleteTrustedDomain – Deletes a trusted domain on the system.

LsaEnumerateAccountRights – Enumerates privileges for a specific account.

LsaEnumerateAccounts – Enumerates what accounts are created on the system.

LsaEnumerateAccountsWithUserRight – Searches for accounts that have privileges.

LsaEnumeratePrivileges – Enumerates privileges for a specific account.

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

Many of these strings can be translated using Google Translate. They are in a few 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

Dynamic Analysis is a technique used by malware analysts to understand the behavior of malware while it is running on a computer system. It needs to be run in a safe environment such as a virtual machine that is isolated from any other networks. Many vendors offer virtualized environments for dynamic analysis, such as Any.Run and JoeSandbox. Virtual machines can also be created in a hypervisor such as VMware or VirtualBox to provide a safe area for dynamic analysis of malicious software.

To begin this section, 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 malware will be run 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 to understand network communication conducted by the malware.

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. Debugging is a process used by malware analysts to understand the functionality of malware: this thread set renders the process. Upon execution, Health-Ebook.exe spawns a process named after itself: Health-Ebook.exe. This second process contains a PID of 1616. This step 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**.

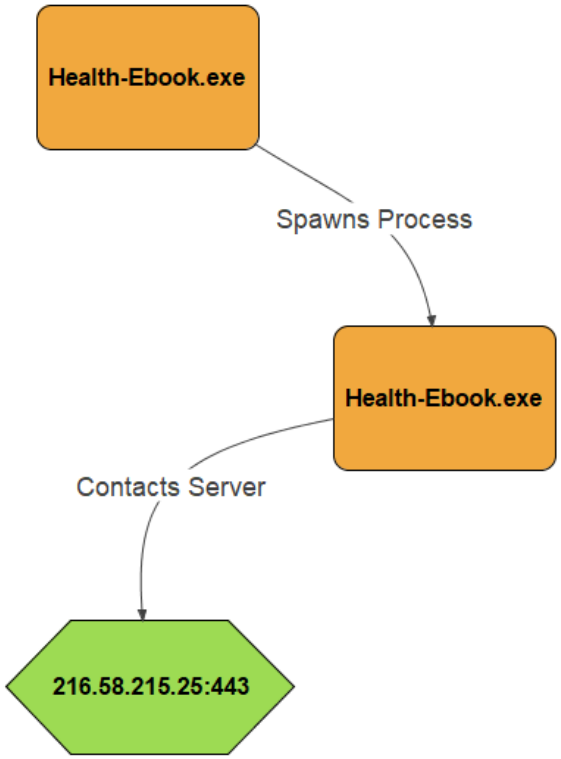


Figure 15: Health-Ebook.exe Process Hollowing and contacting a Google Server

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

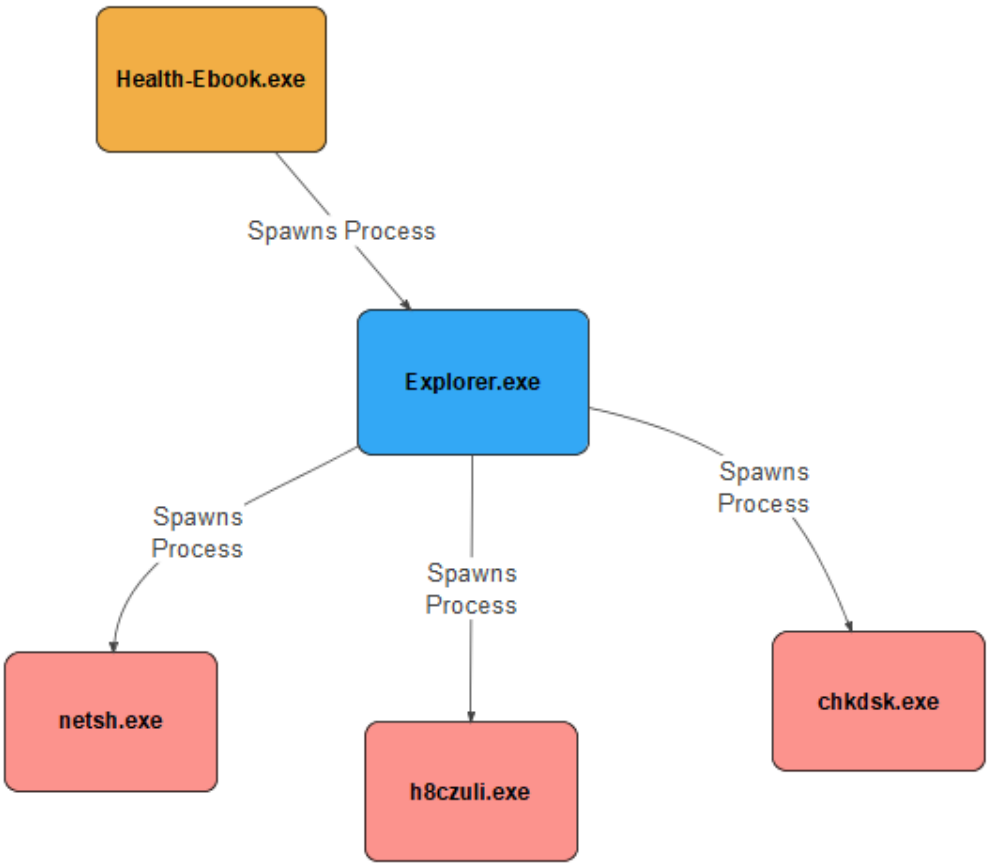


Figure 16: Health-Ebook.exe spawning explorer.exe, which spawns netsh.exe, h8czuli.exe, and chkdsk.exe

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 has the characteristics of Formbook malware. Formbook malware is a type of malware that “steals data, tracks keystrokes, and copies form submissions (hence the "FormBook" name), and it also can receive and execute various commands from the attacker's server” (Miao 2017). 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. Formbook malware is known to store collected data “within the %APPDATA% directory before being sent back to the C&C server” (Swanda 2018). In this example, it copies C:\Users\user\AppData\Local\Google\Chrome\User Data\Default\Login Data and places it in a temporary folder. This is done by one of the **cmd.exe** processes, discussed next. In addition to this Formbook activity, Netsh.exe also creates a registry key to cause a new process, **h8tczuli.exe** run at startup, located in **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**.

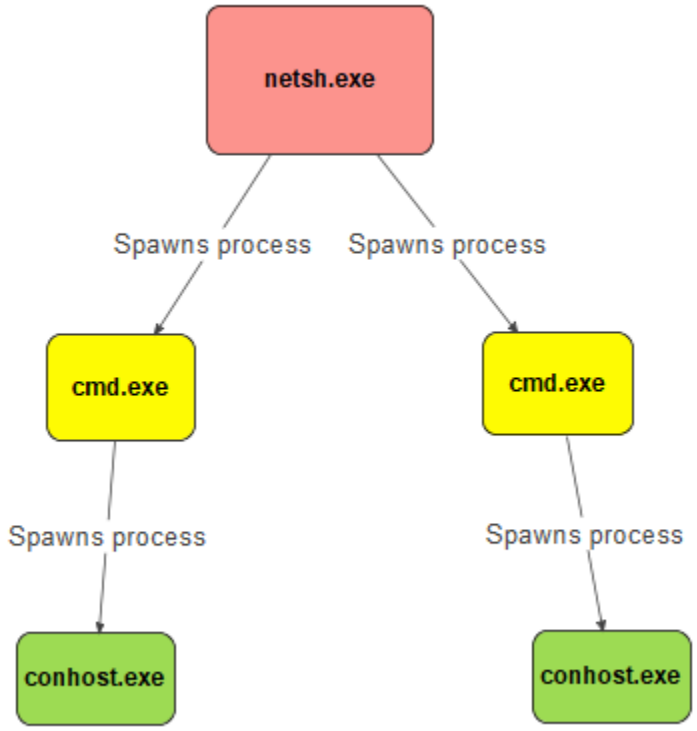


Figure 17: netsh.exe spawning two cmd.exe processes that spawn conhost.exe processes

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. This process could be used to evade detection. Unfortunately, there is a flaw in the code that prevents cmd.exe 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**" thread set. Again, 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)**.

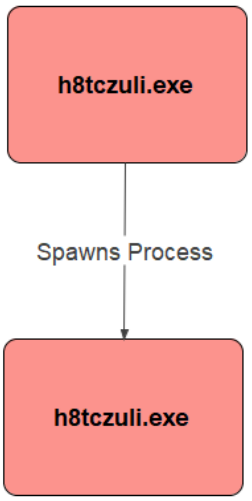


Figure 18: h8tczuli.exe Process Hollowing

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**).

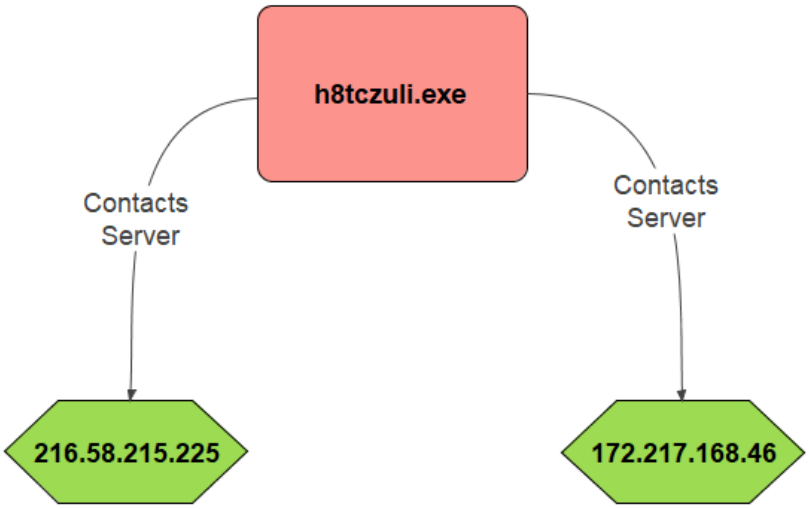


Figure 19: h8czuli.exe Child Process Contacting 2 Google Domains

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 cycle timing in a virtualized environment. By understanding the RDTSC instructions, the malware can determine if it is being executed in a testing or sandbox environment (Oyama 2019).

Figure 20: chkdsk.exe process that intercepts RDTSC instructions to detect a Virtualized Environment

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**). These links appear to be completely legitimate. 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 can be analyzed for malicious indicators. JoeSandbox, an automated dynamic analysis tool, also provides some insight on these IP Addresses.

User Agent

Next, it 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 by blacklisting services 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”. This is obviously the name of a malicious file.

Wireshark .pcap Analysis

Analyzing the Wireshark packet capture provided by the Any.Run dynamic 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 (see **Figure 21**).

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. They're using variations of scripting to make the links unintelligible.

URL Revealer Tool

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

Figure 22: URLRevealer Output in Windows 8 Machine

Both versions of Windows had the same results: connections being made to drive.google.com.

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**).

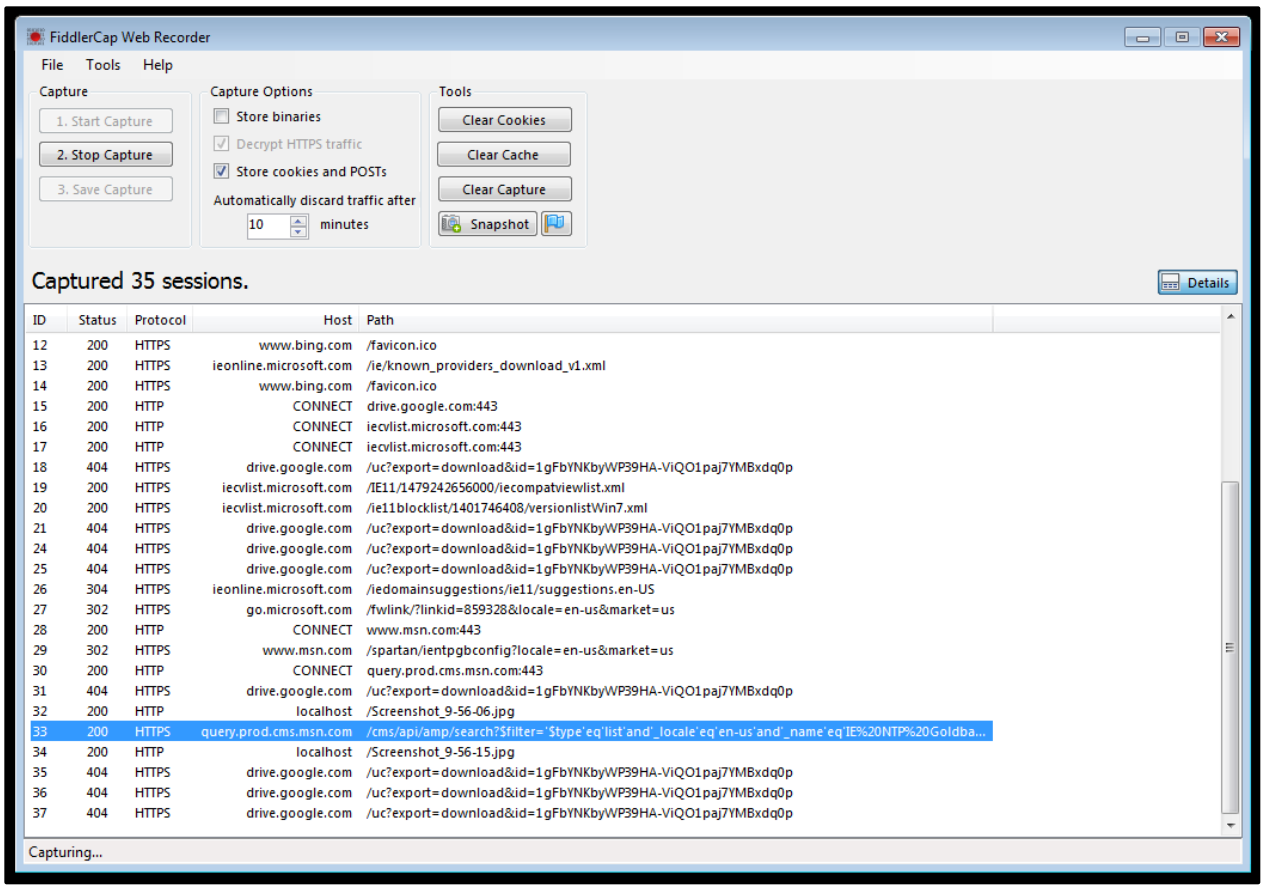


Figure 23: FiddlerCap Web Recorder Output on Windows 8 Machine

The malware appears to be contacting a Google drive URL to download a malicious file. This is the activity of a VB6 GuLoader, which "stores its encrypted payloads on Google Drive" (Proofpoint 2020). The malware can be seen contacting the Google Drive server multiple times during the FiddlerCap Web Recorder capture.

Malware Summary

Looking at the overall activity, this malware resembles a VB5/6 GuLoader. GuLoader malware is used "predominantly to download remote access Trojans (RATs) and information stealers", and in this example, it downloads FormBook malware (Proofpoint 2020). The first few steps after execution are the exact steps that GuLoader will take to evade detection: 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", also known as process hollowing (Proofpoint 2020). The encrypted payload is downloaded from Google Drive, unpacked, and executed. This process makes GuLoader very difficult to analyze, but dynamic analysis can be used to understand its behavior. The downloaded 'Formbook' malware steals login data from web browsers installed on the system (Google Chrome, Internet Explorer, Firefox).

Indicator of Compromise (IOC)

An Indicator of Compromise (IOC) is a forensic artifact that can be used to determine if a specific type of malicious activity has been conducted on an individual device or a network. They "aid information security and IT professionals in detecting data breaches, malware infections, or other threat activity" (Lord 2018). They may include "the MD5 [hash] of a file, a registry path or something found in process memory" (Gibb 2013). IOCs are used in combination with tools such as FireEye OpenIOC to scan for malicious activity. Below is a list of IOCs that were collected from Health-Ebook.exe:

- drive.google.com/uc?export=download&id=1gFbYNKbyWP39HA-ViQO1paj7YMBxdp0p
- http://www.michalshahar.com/w0k/?r65hj=BN90bfcptvP4SJ&3fct=vO9Vm2RARflm5p1PFXqn6eBrWTFFnunBf6X3DMkFEdmGbjkCk/pABuPtOpuxvLvCis20
- http://www.kbasherphotography.com/w0k
- http://www.kbasherphotography.com/w0k/?r65hj=BN90bfcptvP4SJ&3fct=3PkPLEV8daGFL4/3pxhg1tKv6aVypEBkpsp65f+Yzy4XBcektFNWUD7dAcSGsTOSbbgw
- 1E6BC511824F07C5107CB4A5075A811EB1D28F2916630BF7DB1BB5C1649B0E7D
- DB73126EE8583999B121159E70E634CA23FD012D
- 93FBA794DCB6996185F8E93062C12CD4
- drive.google.com

- ocsp.pki.goog
- doc-0s-5o-docs.googleusercontent.com

YARA Rule

YARA is a "tool aimed at (but not limited to) helping malware researchers to identify and classify malware samples" (YARA 2014). YARA rules can be created to scan a specified file for specific characteristics such as strings or hex values. The rule will contain "a set of strings and a boolean expression which determine its logic" (YARA 2014). This expression will return matches based on the provided strings. Below is a YARA rule used to classify Health-Ebook.exe.

```
Rule Health-Ebook_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=vO9Vm2RARflm5p1PFXqn6eBrWTFFn
unBf6X3DMkFEdmGbjkCk/pABuPtOpuxvLvCis20"
        $c = "http://www.kbasherphotography.com/w0k/"
        $d =
"http://www.kbasherphotography.com/w0k/?r65hj=BN90bfcptvP4SJ&3fct=3PkPLEV8daGFL4/3pxhg1tKv6a
VypEBkpsp65f+Yzy4XBcektFNWUD7dAcSGsTOSbbgw"
        $mz = {4d 5A}

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

Conclusion

In conclusion, phishing campaigns can include many different vectors of attack. They can involve simple scams involving emails, pretending to be a trusted entity, malicious websites phishing for credentials or personally identifiable information, or complex malware that can perform malicious functions. The attacks may be themed to attract victims in some way or may present a sense of urgency. The malware sent out in email campaigns is not limited to what it can do: it could be Formbook credential stealing malware, GuLoader malware that utilizes cloud services to download its payload, ransomware, and other forms of malicious software. The malware may have different goals and may steal specific data. It also may target a specific region or specific members of an organization. The malware may download other types of malware or serve ransomware that encrypts the victim's files. If one attack does not work, the attacker may create a more complex method of attack. It is essential to understand what these types attacks look like, so they can be prevented in the future. There are many indicators that a webpage or email are malicious or fake, such as misspellings and broken links.

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

Gibb, W., & Kerr, D. (2013, October 1). OpenIOC: Back to the Basics. Retrieved from https://www.fireeye.com/blog/threat-research/2013/10/openioc-basics.html

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

Miao, Y. (2017) FormBook Is the Latest Example of Malware-as-a-Service. Retrieved from https://www.opswat.com/blog/formbook-latest-example-malware-service

Miller, M. (2008). Is It Safe? Protecting Your Computer, Your Business, and Yourself Online. (p. 127).

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/

Oyama Y. (2019) How Does Malware Use RDTSC? A Study on Operations Executed by Malware with CPU Cycle Measurement. In: Perdisci R., Maurice C., Giacinto G., Almgren M. (eds) Detection of Intrusions and Malware, and Vulnerability Assessment. DIMVA 2019. Lecture Notes in Computer Science, vol 11543. Springer, Cham

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#

Stutz, Michael (January 29, 1998). "AOL: A Cracker's Momma!". Wired News. Archived from the original on December 14, 2005.

Swanda, A. (2018) FormBook Stealer: Data theft made easy. Retrieved from https://inquest.net/blog/2018/06/22/a-look-at-formbook-stealer

YARA (2014) Welcome to YARA's documentation!. Retrieved from https://yara.readthedocs.io/en/stable/