IDENTIFYING MALWARE ATTACKS USING WIRESHARK

Overview:

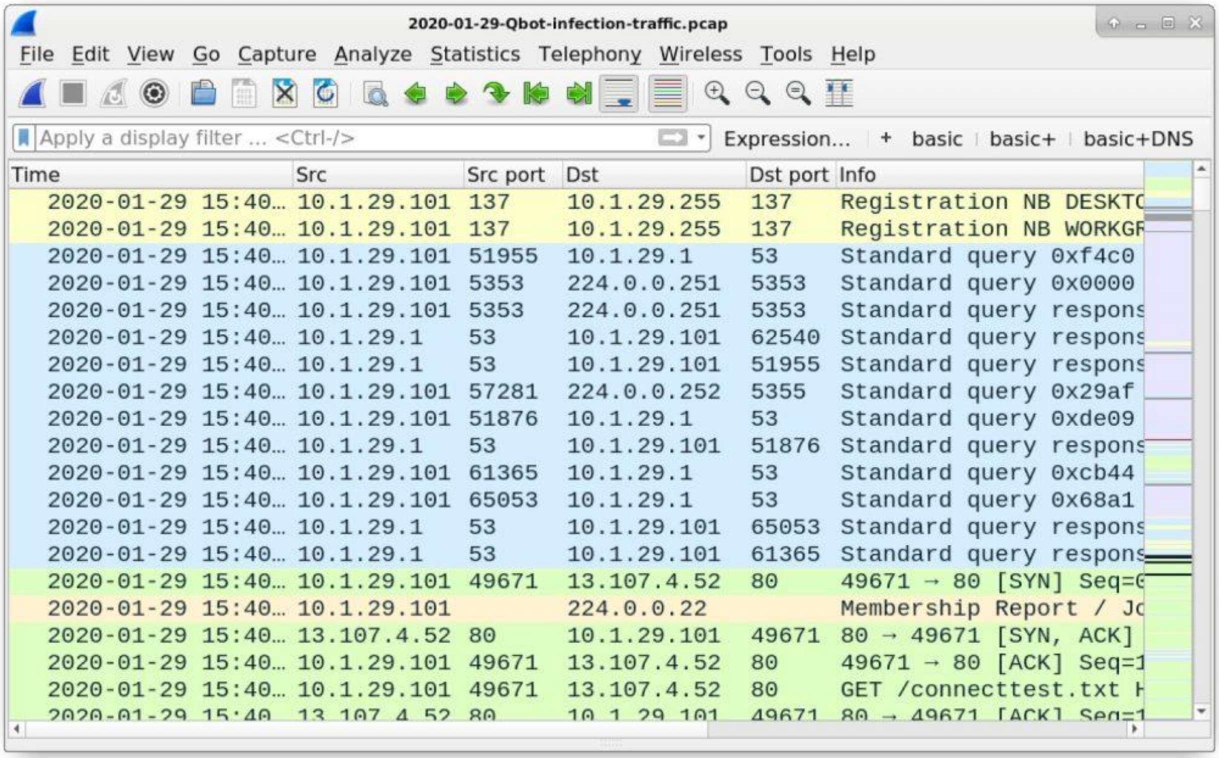
Qakbot is an information stealer also known as Qbot. This family of malware has been active for years, and Qakbot generates distinct traffic patterns.
This Wireshark tutorial reviews a recent packet capture (pcap) from a Qakbot infection. Understanding these traffic patterns can be critical for security professionals when detecting and investigating Qakbot infections.

Note: This tutorial assumes you have a basic knowledge of network traffic and Wireshark. We use a customized column display shown in this tutorial. You should also have experience with Wireshark display filters as described in this additional tutorial.

Please also note that the pcap used for this tutorial contains malware. You should review this pcap in a non-Windows environment. If you are limited to a Windows computer, we suggest reviewing the pcap within a virtual machine (VM) running any of the popular recent Linux distros.

This tutorial will cover the following:

- Qakbot distribution methods
- Initial zip archive from link in an malspam
- Windows executable for Qakbot
- Post-infection HTTPS activity
- Other post-infection traffic


2020-01-29-Qbot-infection-traffic.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+ basic
basic+
basic+DNS

Time	Src	Src port	Dst	Dst port	Info
2020-01-29 15:40…	10.1.29.101	137	10.1.29.255	137	Registration NB DESKTC
2020-01-29 15:40…	10.1.29.101	137	10.1.29.255	137	Registration NB WORKGF
2020-01-29 15:40…	10.1.29.101	51955	10.1.29.1	53	Standard query 0xf4c0
2020-01-29 15:40…	10.1.29.101	5353	224.0.0.251	5353	Standard query 0x0000
2020-01-29 15:40…	10.1.29.101	5353	224.0.0.251	5353	Standard query respons
2020-01-29 15:40…	10.1.29.1	53	10.1.29.101	62540	Standard query respons
2020-01-29 15:40…	10.1.29.1	53	10.1.29.101	51955	Standard query respons
2020-01-29 15:40…	10.1.29.101	57281	224.0.0.252	5355	Standard query 0x29af
2020-01-29 15:40…	10.1.29.101	51876	10.1.29.1	53	Standard query 0xde09
2020-01-29 15:40…	10.1.29.1	53	10.1.29.101	51876	Standard query respons
2020-01-29 15:40…	10.1.29.101	61365	10.1.29.1	53	Standard query 0xcb44
2020-01-29 15:40…	10.1.29.101	65053	10.1.29.1	53	Standard query 0x68a1
2020-01-29 15:40…	10.1.29.1	53	10.1.29.101	65053	Standard query respons
2020-01-29 15:40…	10.1.29.1	53	10.1.29.101	61365	Standard query respons
2020-01-29 15:40…	10.1.29.101	49671	13.107.4.52	80	49671 → 80 [SYN] Seq=0
2020-01-29 15:40…	10.1.29.101		224.0.0.22		Membership Report / Jo
2020-01-29 15:40…	13.107.4.52	80	10.1.29.101	49671	80 → 49671 [SYN, ACK]
2020-01-29 15:40…	10.1.29.101	49671	13.107.4.52	80	49671 → 80 [ACK] Seq=1
2020-01-29 15:40…	10.1.29.101	49671	13.107.4.52	80	GET /connecttest.txt H
2020-01-29 15:40	13.107.4.52	80	10.1.29.101	49671	80 → 49671 [ACK] Seq=1

Figure 1. The pcap for this tutorial.

Qakbot Distribution Methods:

Qakbot is most often distributed through malicious spam (malspam), but it also has been distributed through exploit kits as recently as November 2019. In some cases, Qakbot is a follow-up infection caused by different malware like Emotet as reported in this example from March 2019.

Recent malspam-based distribution campaigns for Qakbot follow a chain of events shown in Figure 2.

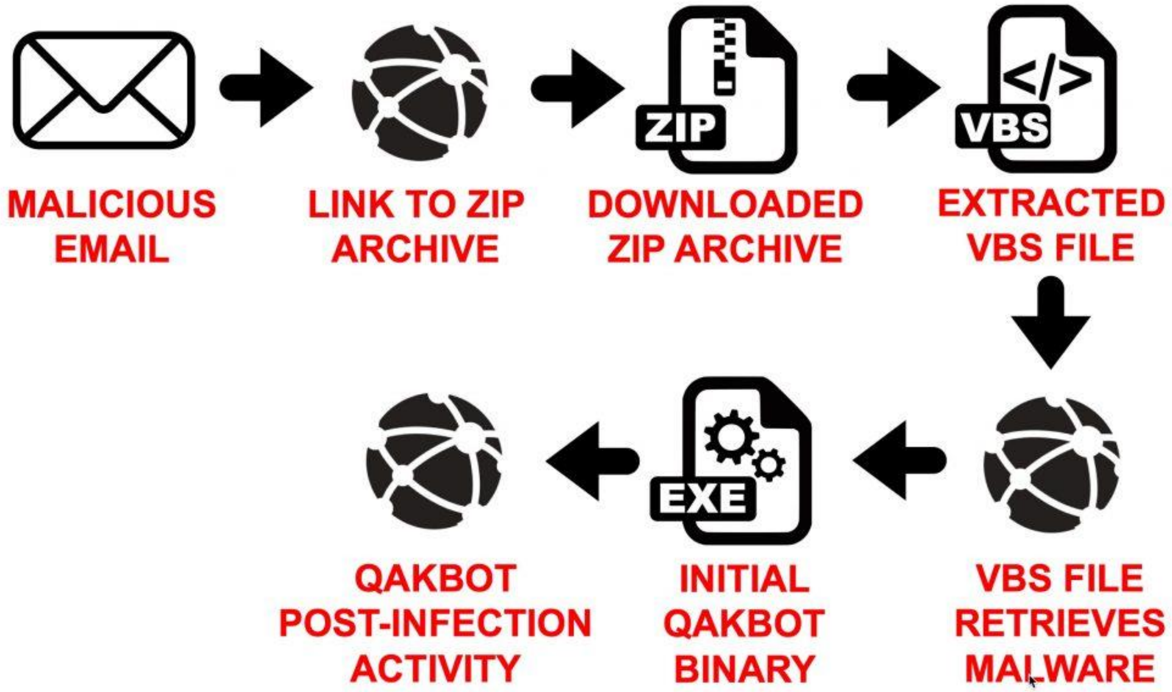
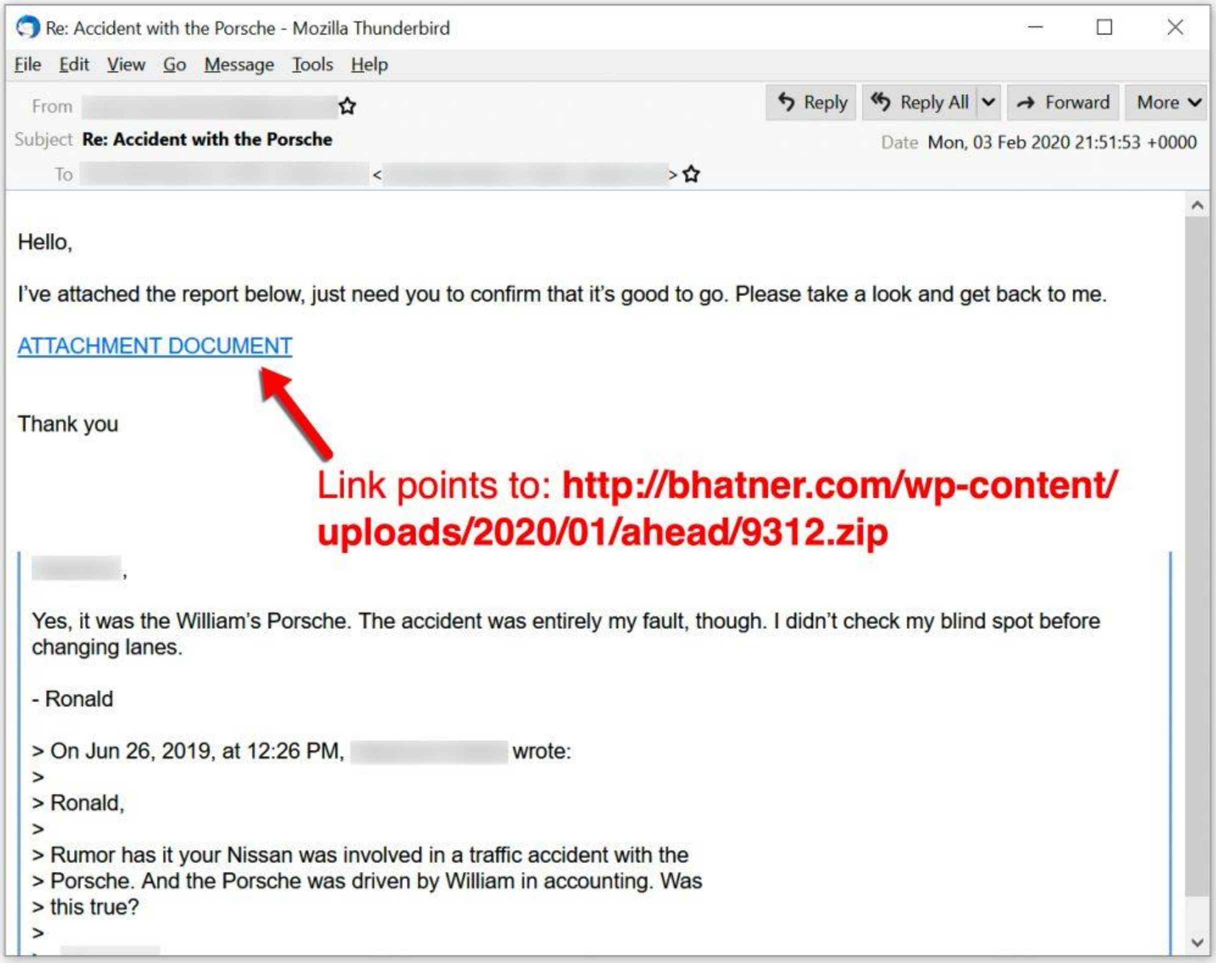


Figure 2. Flow chart from recent Qakbot distribution campaigns.

Initial Zip Archive from Link in Malspam

Recent malspam distributing Qakbot uses fake email chains that spoof legitimate email addresses. One such example is shown in Figure 3.


Re: Accident with the Porsche - Mozilla Thunderbird
File Edit View Go Message Tools Help
Reply
Reply All
Forward
More

From [redacted]

Subject **Re: Accident with the Porsche**

Date Mon, 03 Feb 2020 21:51:53 +0000

To [redacted] <[redacted]>

Hello,

I've attached the report below, just need you to confirm that it's good to go. Please take a look and get back to me.

ATTACHMENT DOCUMENT

Thank you

Link points to: **http://bhatner.com/wp-content/uploads/2020/01/ahead/9312.zip**

> [redacted],
>
> Yes, it was the William's Porsche. The accident was entirely my fault, though. I didn't check my blind spot before changing lanes.
>
> - Ronald
>
> \> On Jun 26, 2019, at 12:26 PM, [redacted] wrote:
> \>
> \> Ronald,
> \>
> \> Rumor has it your Nissan was involved in a traffic accident with the
> \> Porsche. And the Porsche was driven by William in accounting. Was
> \> this true?
> \>

Figure 3. Recent malspam example pushing Qakbot.

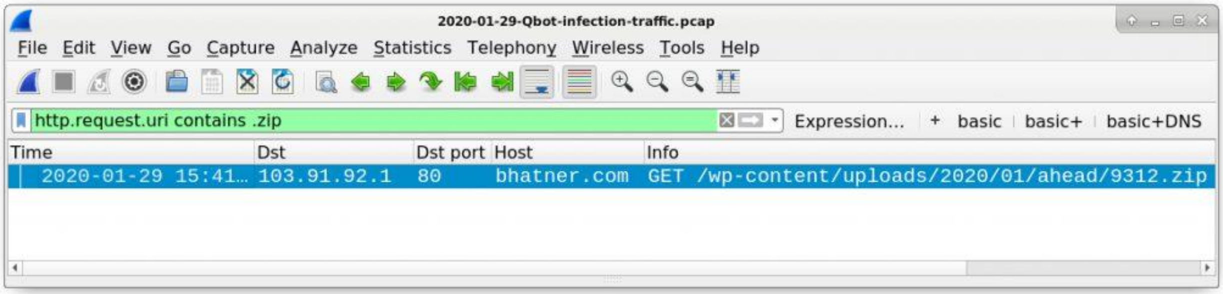
URLs from these emails end with a short series of numbers followed by ***.zip***. See Table 1 for a few examples of URLs from Qakbot malspam recently reported on URLhaus and Twitter.

First reported	**URL for initial zip archive**
2019-12-27	hxxps://prajoon.000webhostapp[.]com/wp-content/uploads/2019/12/last/033/033.zip
2019-12-27	hxxps://psi-uae[.]com/wp-content/uploads/2019/12/last/870853.zip
2019-12-27	hxxps://re365[.]com/wp-content/uploads/2019/12/last/85944289/85944289.zip
2019-12-27	hxxps://liputanforex.web[.]id/wp-content/uploads/2019/12/last/794/794.zip

2020-01-06	hxxp://eps.icothanglong.edu[.]vn/forward/13078.zip
2020-01-22	hxxp://hitechrobo[.]com/wp-content/uploads/2020/01/ahead/84296848/84296848.zip
2020-01-22	hxxp://faithoasis.000webhostapp.com/wp-content/uploads/2020/01/ahead/550889.zip
2020-01-27	hxxps://madisonclubbar[.]com/fast/invoice049740.zip
2020-01-29	hxxp://zhinengbao[.]wang/wp-content/uploads/2020/01/lane/00571.zip
2020-01-29	hxxp://bhatner[.]com/wp-content/uploads/2020/01/ahead/9312.zip
2020-02-03	hxxp://santedeplus[.]info/wp-content/uploads/2020/02/ending/1582820/1582820.zip

Table 1. URLs for the initial zip archive to kick off a Qakbot infection chain.

In our pcap, you can find the HTTP request for a zip archive using ***http.request.uri contains .zip*** in the Wireshark filter as shown in Figure 4.

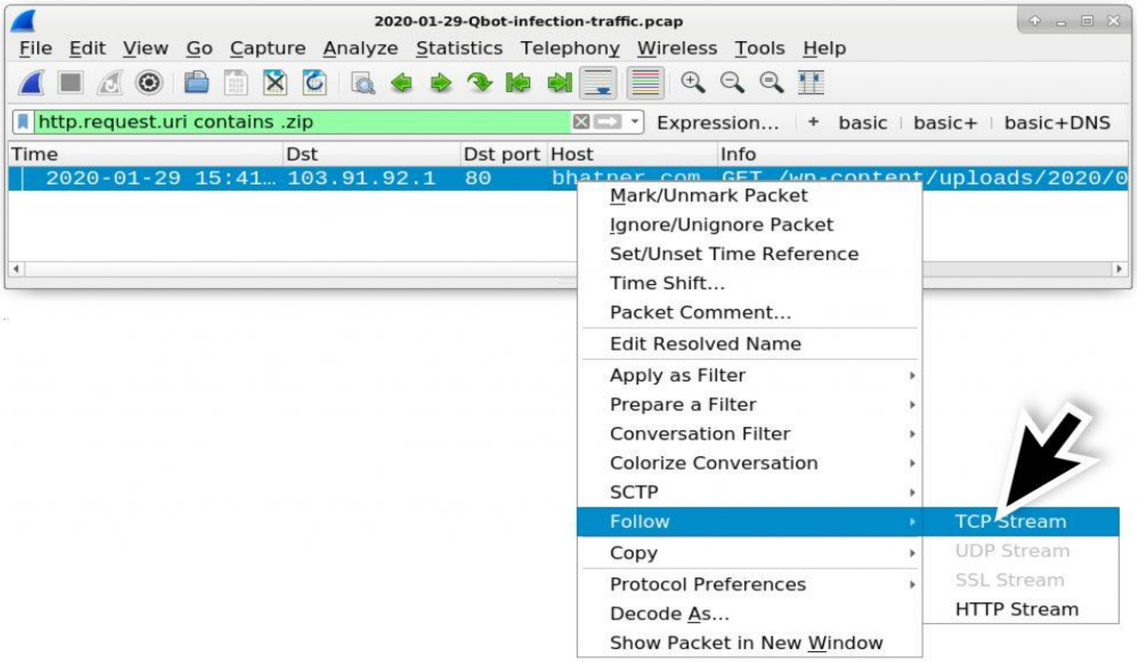

2020-01-29-Qbot-infection-traffic.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...
+ basic
basic+
basic+DNS

http.request.uri contains .zip

Time	Dst	Dst port	Host	Info
2020-01-29 15:41…	103.91.92.1	80	bhatner.com	GET /wp-content/uploads/2020/01/ahead/9312.zip

Figure 4. Finding the URL for the initial zip archive.

Follow the TCP stream to confirm this is a zip archive as shown in Figure 5 and Figure 6, then try to export the zip archive from the pcap as shown in Figure 7.

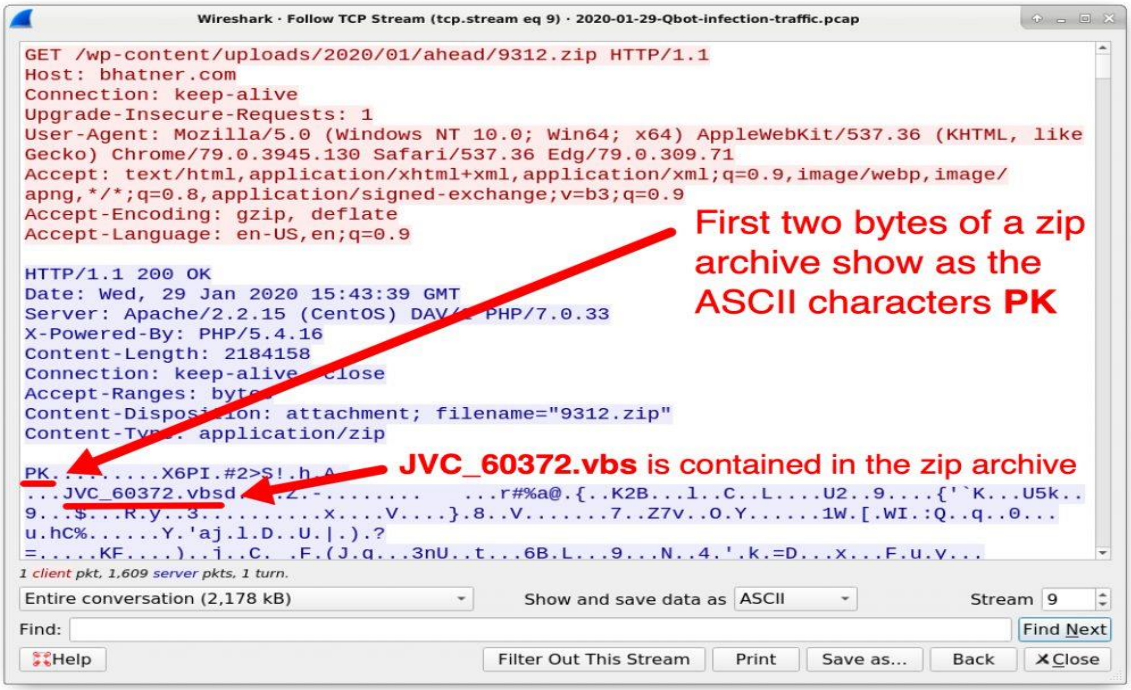


2020-01-29-Qbot-infection-traffic.pcap

http.request.uri contains .zip

```
Time                 Dst          Dst port Host         Info
2020-01-29 15:41…    103.91.92.1  80       bhatner.com  GET /wp-content/uploads/2020/0
```

Figure 5. Following the TCP stream for the HTTP request from our filter results.



Wireshark · Follow TCP Stream (tcp.stream eq 9) · 2020-01-29-Qbot-infection-traffic.pcap

```
GET /wp-content/uploads/2020/01/ahead/9312.zip HTTP/1.1
Host: bhatner.com
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/79.0.3945.130 Safari/537.36 Edg/79.0.309.71
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/
apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

HTTP/1.1 200 OK
Date: Wed, 29 Jan 2020 15:43:39 GMT
Server: Apache/2.2.15 (CentOS) DAV/2 PHP/7.0.33
X-Powered-By: PHP/5.4.16
Content-Length: 2184158
Connection: keep-alive, close
Accept-Ranges: bytes
Content-Disposition: attachment; filename="9312.zip"
Content-Type: application/zip

PK.........X6PI.#2>S!.h.A
...JVC_60372.vbsd..Z.-........    ...r#%a@.{..K2B...l..C..L....U2..9....{'`K...U5k..
9...$...R.y..3.........x....V....}.8..V.......7..Z7v..O.Y......1W.[.WI.:Q..q..0...
u.hC%......Y.'aj.l.D..U.|.).?
=.....KF....)..j..C. .F.(J.q...3nU..t...6B.L...9...N..4.'.k.=D...x...F.u.v...
```

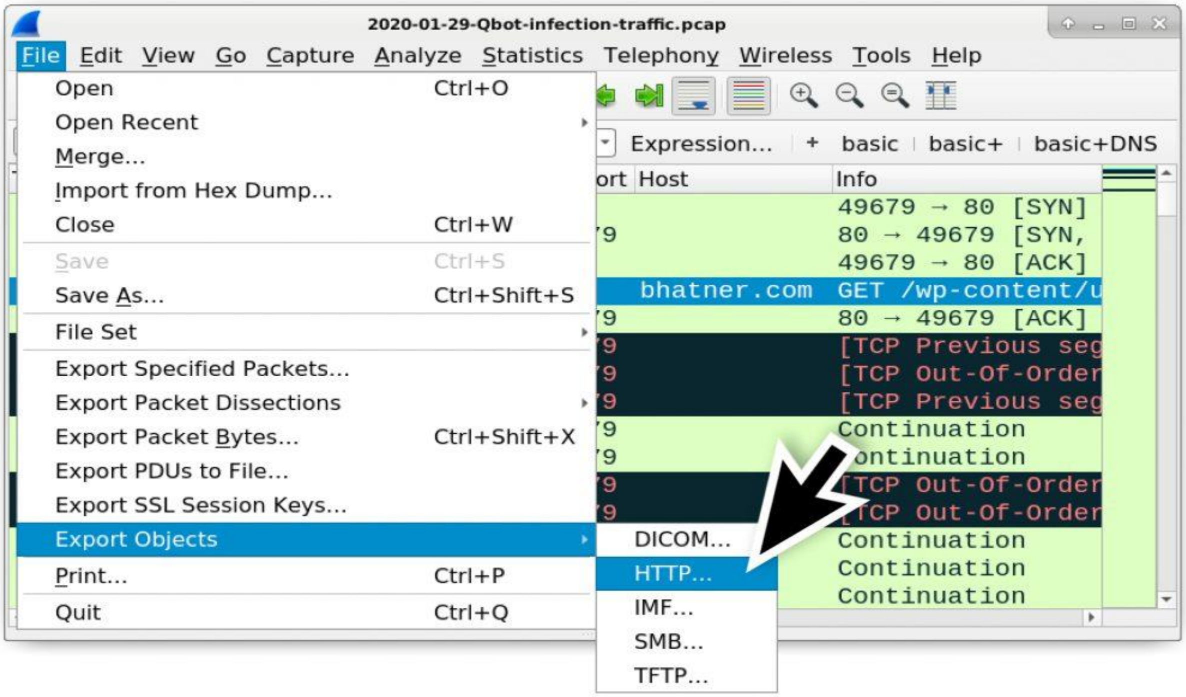
1 client pkt, 1,609 server pkts, 1 turn.

Entire conversation (2,178 kB) Show and save data as ASCII Stream 9

Find:

Find Next
Help
Filter Out This Stream
Print
Save as…
Back
Close
First two bytes of a zip archive show as the ASCII characters PK
JVC_60372.vbs is contained in the zip archive

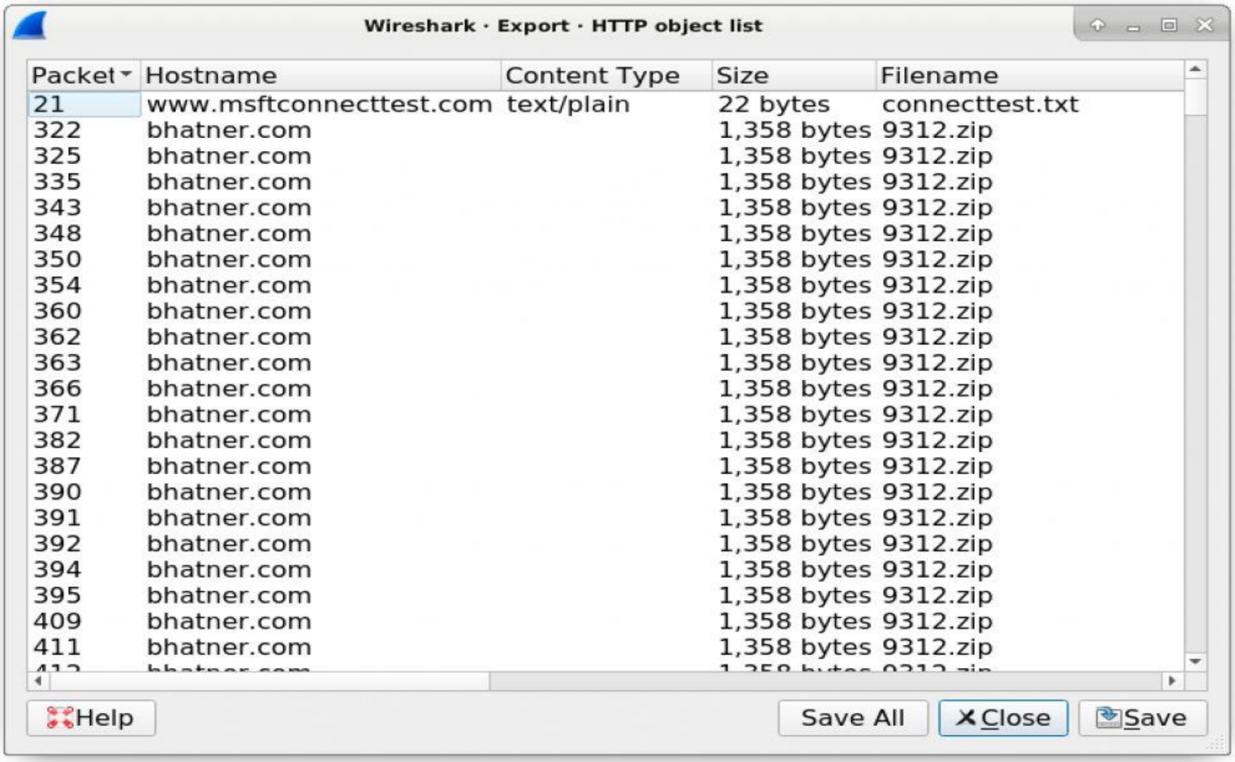
Figure 6. Indicators this URL returned a zip archive



```
                    49679 → 80 [SYN]
'9                  80 → 49679 [SYN,
                    49679 → 80 [ACK]
   bhatner.com      GET /wp-content/u
'9                  80 → 49679 [ACK]
'9                  [TCP Previous seg
'9                  [TCP Out-Of-Order
'9                  [TCP Previous seg
'9                  Continuation
'9                  ontinuation
'9                  [TCP Out-Of-Order
'9                  TCP Out-Of-Order
                    Continuation
                    Continuation
                    Continuation
```

Figure 7. Exporting objects from HTTP traffic in the pcap.

In most cases, the menu for ***File → Export Objects → HTTP*** should export a zip archive sent over HTTP. Unfortunately, as shown in Figure 8, we cannot export this file named ***9312.zip*** because it is separated into hundreds of smaller parts within the export HTTP objects list.



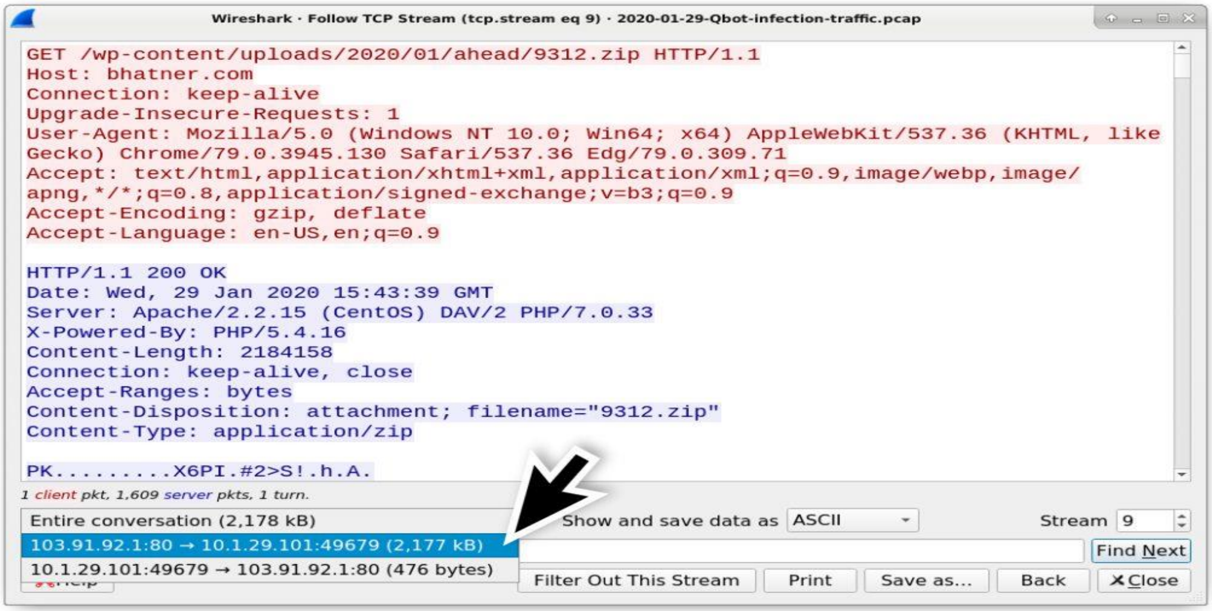
Packet	Hostname	Content Type	Size	Filename
21	www.msftconnecttest.com	text/plain	22 bytes	connecttest.txt
322	bhatner.com		1,358 bytes	9312.zip
325	bhatner.com		1,358 bytes	9312.zip
335	bhatner.com		1,358 bytes	9312.zip
343	bhatner.com		1,358 bytes	9312.zip
348	bhatner.com		1,358 bytes	9312.zip
350	bhatner.com		1,358 bytes	9312.zip
354	bhatner.com		1,358 bytes	9312.zip
360	bhatner.com		1,358 bytes	9312.zip
362	bhatner.com		1,358 bytes	9312.zip
363	bhatner.com		1,358 bytes	9312.zip
366	bhatner.com		1,358 bytes	9312.zip
371	bhatner.com		1,358 bytes	9312.zip
382	bhatner.com		1,358 bytes	9312.zip
387	bhatner.com		1,358 bytes	9312.zip
390	bhatner.com		1,358 bytes	9312.zip
391	bhatner.com		1,358 bytes	9312.zip
392	bhatner.com		1,358 bytes	9312.zip
394	bhatner.com		1,358 bytes	9312.zip
395	bhatner.com		1,358 bytes	9312.zip
409	bhatner.com		1,358 bytes	9312.zip
411	bhatner.com		1,358 bytes	9312.zip
413	bhatner.com		1,358 bytes	9312.zip

Figure 8. 9312.zip is broken up into hundreds of objects within the list, so we cannot export it this way.

Fortunately, we can export data from a TCP stream window and edit the binary in a hex editor to remove any hxxP response headers. Use the following steps to extract the zip archive from this pcap:

1. Follow TCP stream for the HTTP request for 9312.zip.
2. Show only the response traffic in the TCP stream Window.
3. Change “Show and save data as” from ASCII to Raw.
4. Save the data as a binary (I chose to save it as: 9312.zip.bin)
5. Open the binary in a hex editor and remove the HTTP request headers before the first two bytes of the zip archive (which show as PK in ASCII).
6. Save the file as a zip archive (I chose to save it as 9312.zip)
7. Check the file to make sure it’s a zip archive.

See Figures 9 through 14 for a visual guide of this process.



```
GET /wp-content/uploads/2020/01/ahead/9312.zip HTTP/1.1
Host: bhatner.com
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/79.0.3945.130 Safari/537.36 Edg/79.0.309.71
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/
apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

HTTP/1.1 200 OK
Date: Wed, 29 Jan 2020 15:43:39 GMT
Server: Apache/2.2.15 (CentOS) DAV/2 PHP/7.0.33
X-Powered-By: PHP/5.4.16
Content-Length: 2184158
Connection: keep-alive, close
Accept-Ranges: bytes
Content-Disposition: attachment; filename="9312.zip"
Content-Type: application/zip

PK.........X6PI.#2>S!.h.A.
```

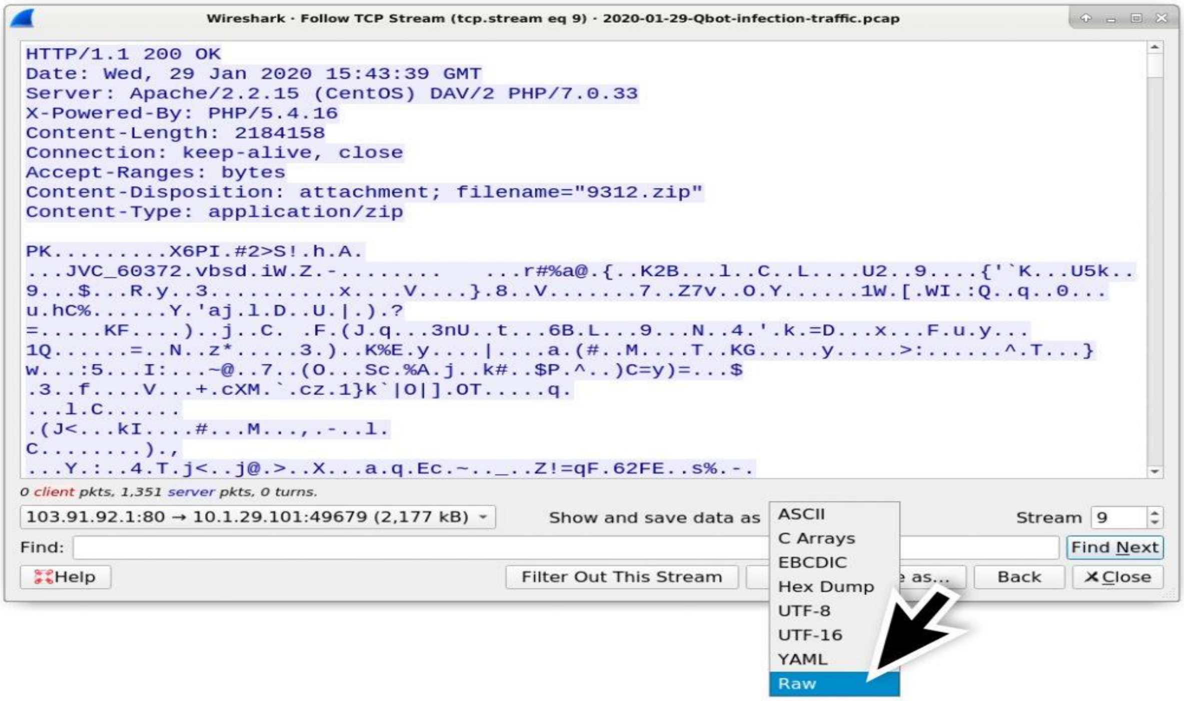
1 client pkt, 1,609 server pkts, 1 turn.

Show and save data as: ASCII

Stream: 9

Wireshark · Follow TCP Stream (tcp.stream eq 9) · 2020-01-29-Qbot-infection-traffic.pcap
Entire conversation (2,178 kB)
103.91.92.1:80 → 10.1.29.101:49679 (2,177 kB)
10.1.29.101:49679 → 103.91.92.1:80 (476 bytes)
Find Next
Filter Out This Stream
Print
Save as…
Back
Close

Figure 9. Step 2 – When viewing the TCP stream, switch from viewing the entire conversation to viewing only data returned from the server.



```
HTTP/1.1 200 OK
Date: Wed, 29 Jan 2020 15:43:39 GMT
Server: Apache/2.2.15 (CentOS) DAV/2 PHP/7.0.33
X-Powered-By: PHP/5.4.16
Content-Length: 2184158
Connection: keep-alive, close
Accept-Ranges: bytes
Content-Disposition: attachment; filename="9312.zip"
Content-Type: application/zip

PK.........X6PI.#2>S!.h.A.
...JVC_60372.vbsd.iW.Z.-........    ...r#%a@.{..K2B...l..C..L....U2..9....{'`K...U5k..
9...$...R.y..3.........x....V....}.8..V.......7..Z7v..O.Y......1W.[.WI.:Q..q..0...
u.hC%......Y.'aj.l.D..U.|.).?
=.....KF....)..j..C. .F.(J.q...3nU..t...6B.L...9...N..4.'.k.=D...x...F.u.y...
1Q......=..N..z*.....3.)..K%E.y....|....a.(#..M....T..KG.....y.....>:......^.T...}
w...:5...I:...~@..7..(O...Sc.%A.j..k#..$P.^..)C=y)=...$
.3..f....V...+.cXM.`.cz.1}k`|O|].OT.....q.
...l.C......
.(J<...kI....#...M...,.-..l.
C........).,
...Y.:..4.T.j<..j@.>..X...a.q.Ec.~.._..Z!=qF.62FE..s%.-.
```

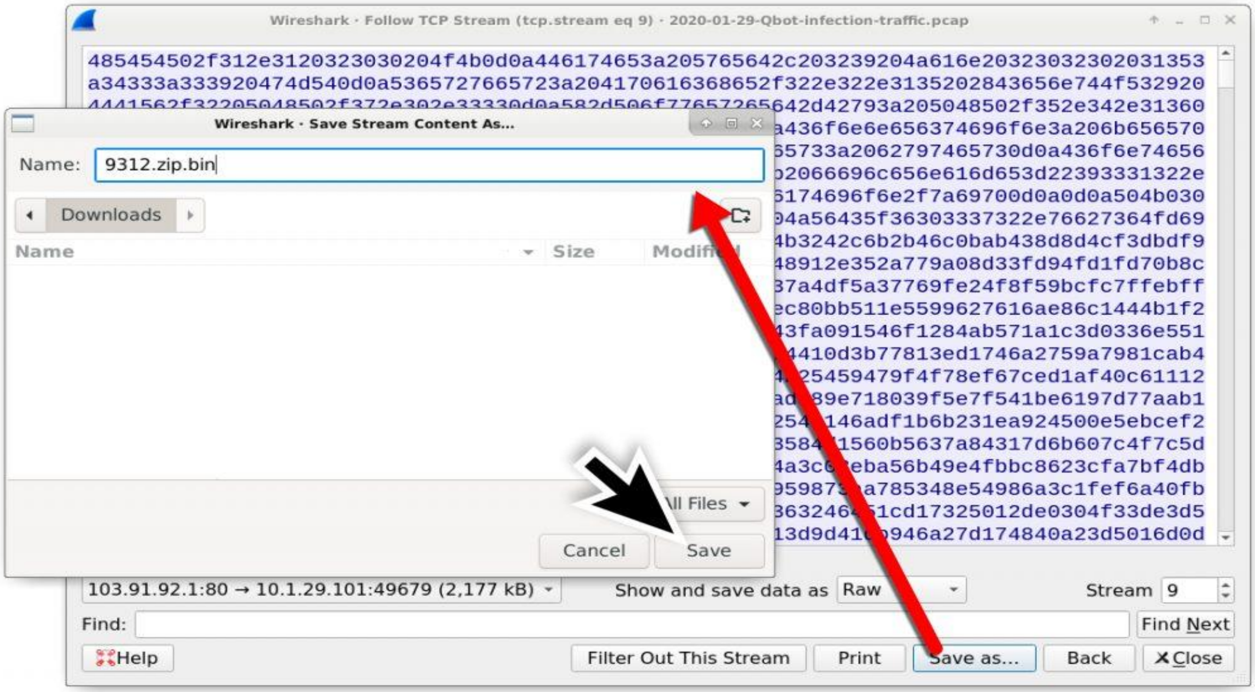
0 client pkts, 1,351 server pkts, 0 turns.

103.91.92.1:80 → 10.1.29.101:49679 (2,177 kB)

Show and save data as: Raw

Stream: 9

Figure 10. Step 3 – Show and save data as Raw instead of ASCII.



Wireshark · Follow TCP Stream (tcp.stream eq 9) · 2020-01-29-Qbot-infection-traffic.pcap

```
485454502f312e3120323030204f4b0d0a446174653a205765642c203239204a616e2032303230203135З
a34333a333920474d540d0a5365727665723a204170616368652f322e322e3135202843656e744f532920
4441562f32205048502f372e302e33330d0a582d506f77657265642d42793a205048502f352e342e31360
                                                           a436f6e6e656374696f6e3a206b656570
                                                           65733a2062797465730d0a436f6e74656
                                                           b2066696c656e616d653d22393331322e
                                                           6174696f6e2f7a69700d0a0d0a504b030
                                                           94a56435f36303337322e76627364fd69
                                                           4b3242c6b2b46c0bab438d8d4cf3dbdf9
                                                           48912e352a779a08d33fd94fd1fd70b8c
                                                           37a4df5a37769fe24f8f59bcfc7ffebff
                                                           ec80bb511e5599627616ae86c1444b1f2
                                                           43fa091546f1284ab571a1c3d0336e551
                                                            4410d3b77813ed1746a2759a7981cab4
                                                           4 25459479f4f78ef67ced1af40c61112
                                                           ad 89e718039f5e7f541be6197d77aab1
                                                           254 146adf1b6b231ea924500e5ebcef2
                                                           3584 1560b5637a84317d6b607c4f7c5d
                                                           4a3c  eba56b49e4fbbc8623cfa7bf4db
                                                           95987  a785348e54986a3c1fef6a40fb
                                                           363246  1cd17325012de0304f33de3d5
                                                           13d9d41  946a27d174840a23d5016d0d
```

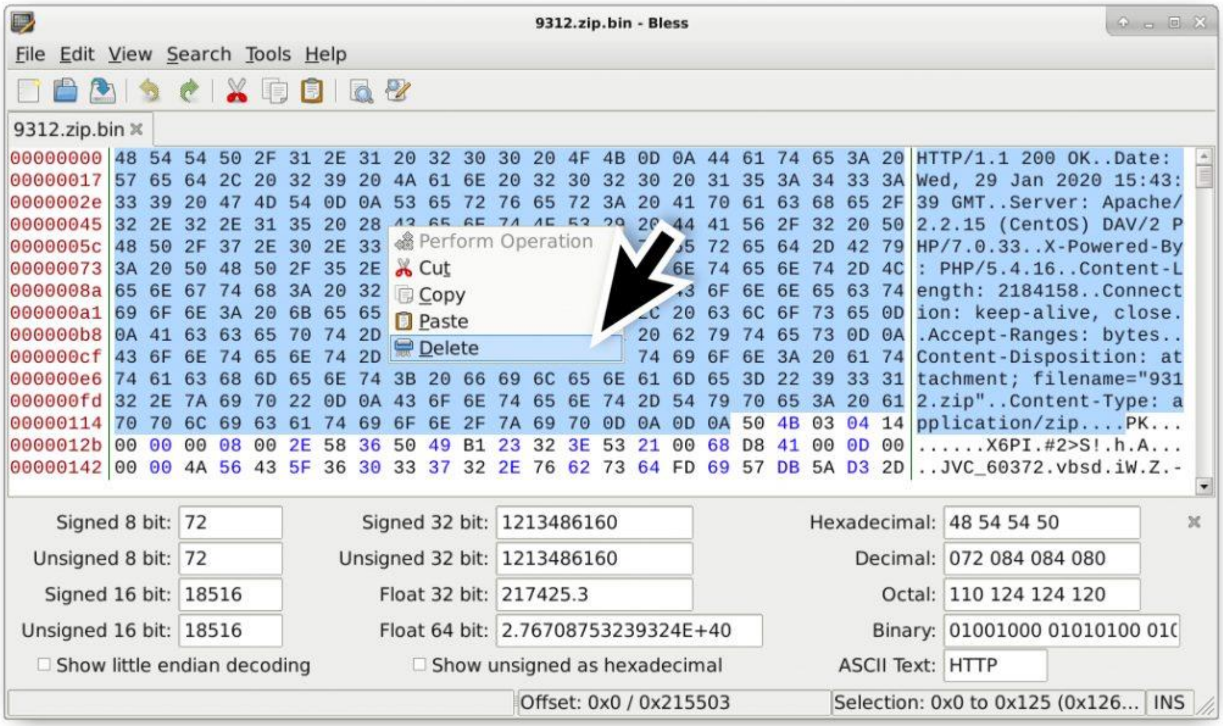
Name: 9312.zip.bin

103.91.92.1:80 → 10.1.29.101:49679 (2,177 kB)

Show and save data as Raw

Stream 9

Figure 11. Step 4 – Save this raw data from the TCP stream as a binary.



```
00000000 48 54 54 50 2F 31 2E 31 20 32 30 30 20 4F 4B 0D 0A 44 61 74 65 3A 20 HTTP/1.1 200 OK..Date:
00000017 57 65 64 2C 20 32 39 20 4A 61 6E 20 32 30 32 30 20 31 35 3A 34 33 3A Wed, 29 Jan 2020 15:43:
0000002e 33 39 20 47 4D 54 0D 0A 53 65 72 76 65 72 3A 20 41 70 61 63 68 65 2F 39 GMT..Server: Apache/
00000045 32 2E 32 2E 31 35 20 28 43 65 6E 74 4F 53 29 20 44 41 56 2F 32 20 50 2.2.15 (CentOS) DAV/2 P
0000005c 48 50 2F 37 2E 30 2E 33                      65 72 65 64 2D 42 79 HP/7.0.33..X-Powered-By
00000073 3A 20 50 48 50 2F 35 2E                   6E 74 65 6E 74 2D 4C : PHP/5.4.16..Content-L
0000008a 65 6E 67 74 68 3A 20 32                      6F 6E 6E 65 63 74 ength: 2184158..Connect
000000a1 69 6F 6E 3A 20 6B 65 65                   2C 20 63 6C 6F 73 65 0D ion: keep-alive, close.
000000b8 0A 41 63 63 65 70 74 2D                   20 62 79 74 65 73 0D 0A .Accept-Ranges: bytes..
000000cf 43 6F 6E 74 65 6E 74 2D                   74 69 6F 6E 3A 20 61 74 Content-Disposition: at
000000e6 74 61 63 68 6D 65 6E 74 3B 20 66 69 6C 65 6E 61 6D 65 3D 22 39 33 31 tachment; filename="931
000000fd 32 2E 7A 69 70 22 0D 0A 43 6F 6E 74 65 6E 74 2D 54 79 70 65 3A 20 61 2.zip"..Content-Type: a
00000114 70 70 6C 69 63 61 74 69 6F 6E 2F 7A 69 70 0D 0A 0D 0A 50 4B 03 04 14 pplication/zip....PK...
0000012b 00 00 00 08 00 2E 58 36 50 49 B1 23 32 3E 53 21 00 68 D8 41 00 0D 00 ......X6PI.#2>S!.h.A...
00000142 00 00 4A 56 43 5F 36 30 33 37 32 2E 76 62 73 64 FD 69 57 DB 5A D3 2D ..JVC_60372.vbsd.iW.Z.-
```

Signed 8 bit: 72

Unsigned 8 bit: 72

Signed 16 bit: 18516

Unsigned 16 bit: 18516

□ Show little endian decoding

Signed 32 bit: 1213486160

Unsigned 32 bit: 1213486160

Float 32 bit: 217425.3

Float 64 bit: 2.76708753239324E+40

□ Show unsigned as hexadecimal

Hexadecimal: 48 54 54 50

Decimal: 072 084 084 080

Octal: 110 124 124 120

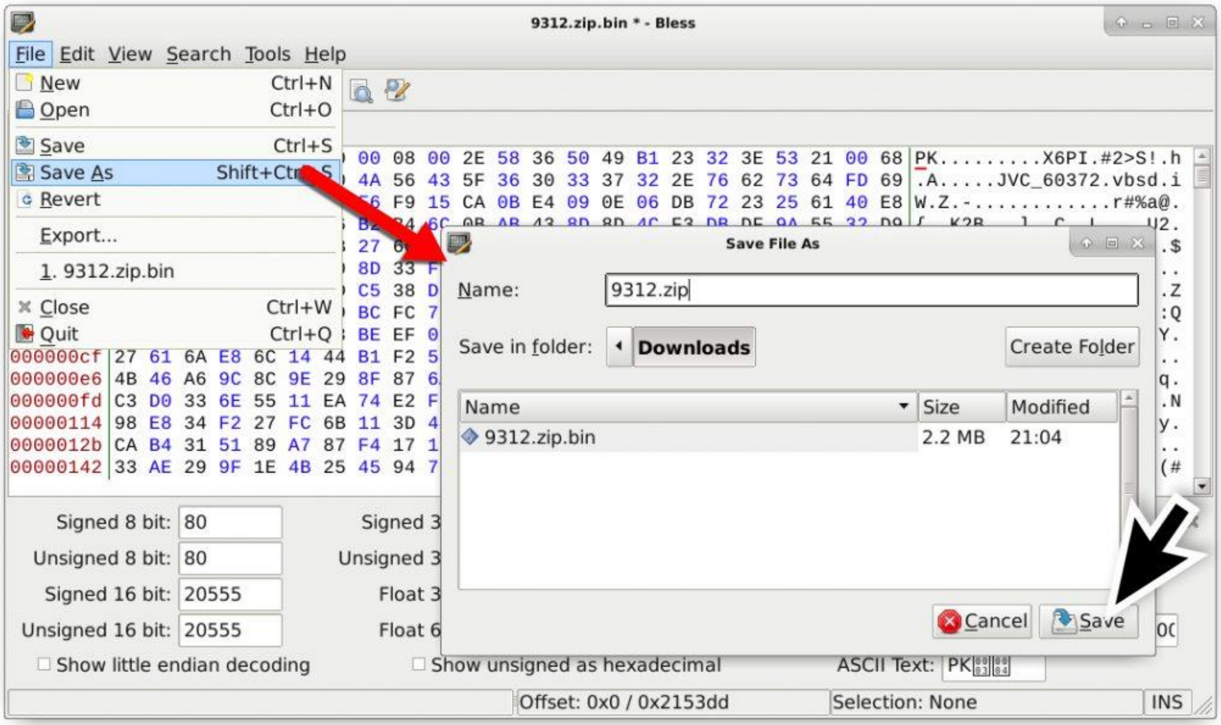
Binary: 01001000 01010100 010

ASCII Text: HTTP

Offset: 0x0 / 0x215503

Selection: 0x0 to 0x125 (0x126... INS

Figure 12. Step 5 – Open your saved binary in a hex editor and remove any HTTP response data before the first two bytes of the zip archive (that show as PK in ASCII).



Name: 9312.zip

Save in folder: Downloads

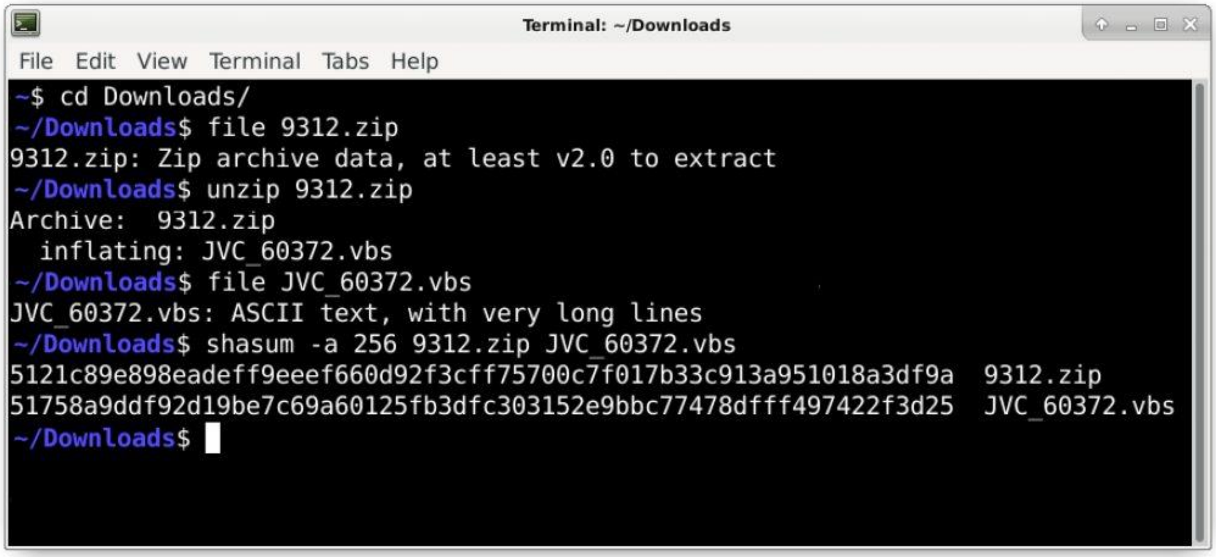
Name	Size	Modified
9312.zip.bin	2.2 MB	21:04

Signed 8 bit: 80
Unsigned 8 bit: 80
Signed 16 bit: 20555
Unsigned 16 bit: 20555

ASCII Text: PK

Offset: 0x0 / 0x2153dd
Selection: None

Figure 13. Step 6 – Save your edited binary as a zip archive.



```
~$ cd Downloads/
~/Downloads$ file 9312.zip
9312.zip: Zip archive data, at least v2.0 to extract
~/Downloads$ unzip 9312.zip
Archive:  9312.zip
  inflating: JVC_60372.vbs
~/Downloads$ file JVC_60372.vbs
JVC_60372.vbs: ASCII text, with very long lines
~/Downloads$ shasum -a 256 9312.zip JVC_60372.vbs
5121c89e898eadeff9eeef660d92f3cff75700c7f017b33c913a951018a3df9a  9312.zip
51758a9ddf92d19be7c69a60125fb3dfc303152e9bbc77478dfff497422f3d25  JVC_60372.vbs
~/Downloads$ 
```

Figure 14. Step 7 – Confirm the edited file is a zip archive, then extract the VBS file and check the file hashes.

Figure 14 shows how to use a terminal window from a Debian-based Linux distro to check the files. From our pcap, the zip archive should be the same as this file submitted to VirusTotal. Our extracted VBS file should be the same as this file also submitted to VirusTotal.

A public sandbox analysis of our extracted VBS file indicates it generates the next Qakbot-related URL in our infection chain: a URL that returned a Windows executable for Qakbot.

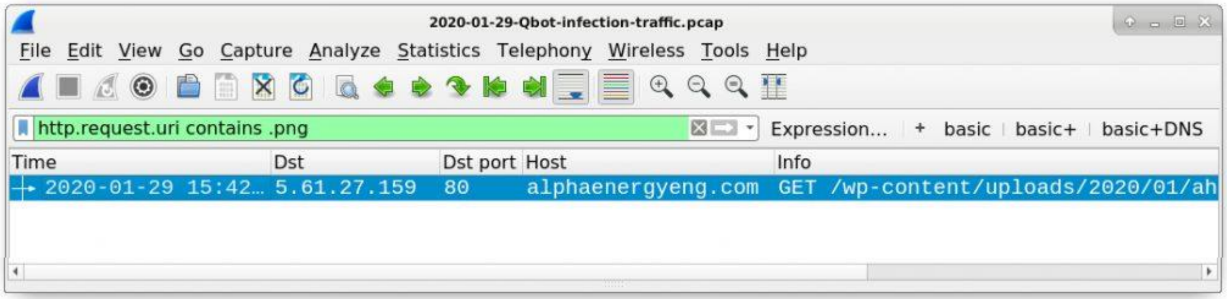
Windows Executable for Qakbot

These extracted VBS files generate URLs that return Windows executables for Qakbot. Since December 2019, URLs for Qakbot executables have ended with 44444.png or 444444.png. See Table 2 for some recent examples of these Qakbot URLs we found using our AutoFocus Threat Intelligence service.

First Seen	**URL for Qakbot executable**
2019-12-27	hxxp://centre-de-conduite-roannais[.]com/wp-content/uploads/2019/12/last/444444.png
2020-01-06	hxxp://newsinside[.]info/wp-content/uploads/2020/01/forward/44444.png
2020-01-15	hxxp://iike.xolva[.]com/wp-content/themes/keenshot/fast/444444.png
2020-01-17	hxxp://deccolab[.]com/fast/444444.png
2020-01-21	hxxp://myrestaurant.coupoly[.]com/wp-content/uploads/2020/01/along/444444.png
2020-01-22	hxxp://alphaenergyeng[.]com/wp-content/uploads/2020/01/ahead/444444.png
2020-01-23	hxxp://claramohammedschoolstl[.]org/wp-content/uploads/2020/01/upwards/444444.png
2020-01-23	hxxp://creationzerodechet[.]com/choice/444444.png
2020-01-26	hxxp://productsphotostudio[.]com/wp-content/uploads/2020/01/lane/444444.png
2020-01-27	hxxp://sophistproduction[.]com/wp-content/uploads/2020/01/choice/444444.png
2020-01-30	hxxp://uofnpress[.]ch/wp-content/uploads/2020/01/side/444444.png
2020-02-03	hxxp://csrkanjiza[.]rs/wp-content/uploads/2020/02/ending/444444.png

Table 2. URLs for Qakbot executables.

In our pcap, find the HTTP GET request for our Qakbot executable using ***hxxp.request.uri contains .png*** in the Wireshark filter as shown in Figure 15.

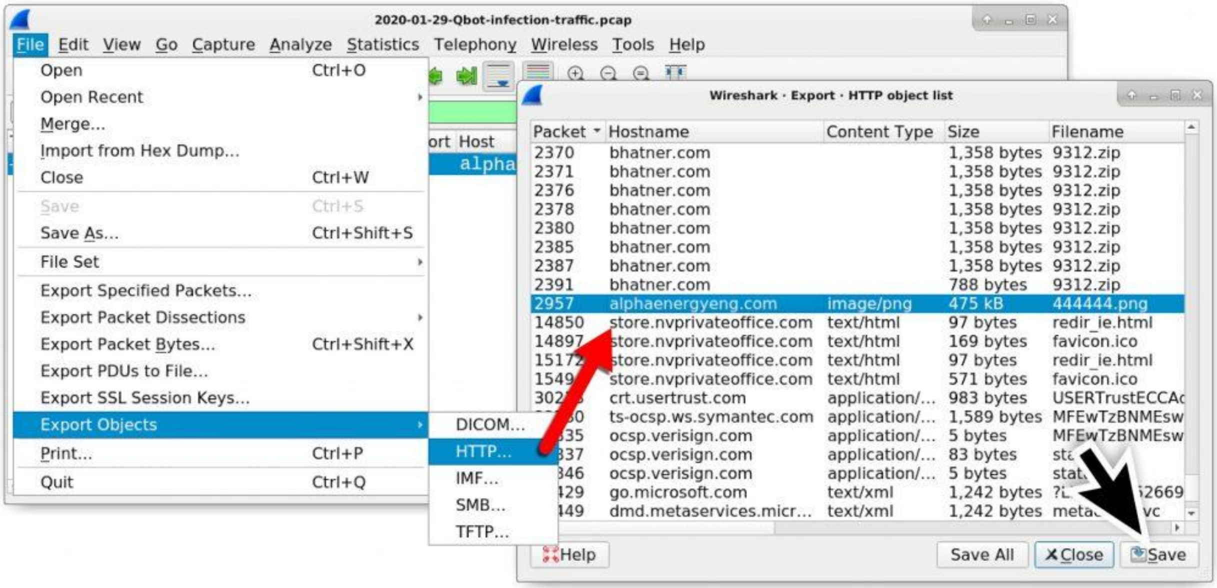

2020-01-29-Qbot-infection-traffic.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression… + basic basic+ basic+DNS

http.request.uri contains .png

Time	Dst	Dst port	Host	Info
2020-01-29 15:42…	5.61.27.159	80	alphaenergyeng.com	GET /wp-content/uploads/2020/01/ah

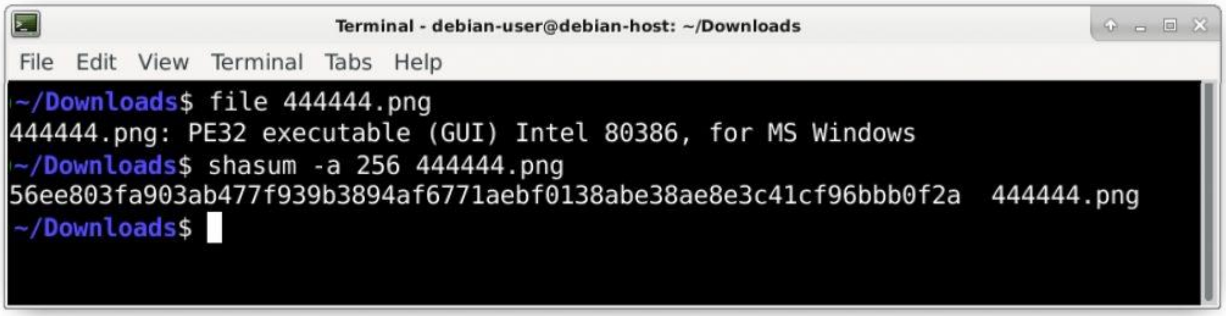
Figure 15. Finding the URL for our Qakbot executable.

Export this object from the pcap using the ***File → Export Objects → HTTP*** menu path as shown in Figure 16 and check the results as shown in Figure 17.



Packet	Hostname	Content Type	Size	Filename
2370	bhatner.com		1,358 bytes	9312.zip
2371	bhatner.com		1,358 bytes	9312.zip
2376	bhatner.com		1,358 bytes	9312.zip
2378	bhatner.com		1,358 bytes	9312.zip
2380	bhatner.com		1,358 bytes	9312.zip
2385	bhatner.com		1,358 bytes	9312.zip
2387	bhatner.com		1,358 bytes	9312.zip
2391	bhatner.com		788 bytes	9312.zip
2957	alphaenergyeng.com	image/png	475 kB	444444.png
14850	store.nvprivateoffice.com	text/html	97 bytes	redir_ie.html
14897	store.nvprivateoffice.com	text/html	169 bytes	favicon.ico
1517	store.nvprivateoffice.com	text/html	97 bytes	redir_ie.html
1549	store.nvprivateoffice.com	text/html	571 bytes	favicon.ico
302	crt.usertrust.com	application/…	983 bytes	USERTrustECCAc
0	ts-ocsp.ws.symantec.com	application/…	1,589 bytes	MFEwTzBNMEsw
35	ocsp.verisign.com	application/…	5 bytes	MFEwTzBNMEsw
337	ocsp.verisign.com	application/…	83 bytes	sta
346	ocsp.verisign.com	application/…	5 bytes	stat
429	go.microsoft.com	text/xml	1,242 bytes	?L 2669
449	dmd.metaservices.micr…	text/xml	1,242 bytes	meta vc

Figure 16. Exporting our Qakbot executable from the pcap.


Terminal - debian-user@debian-host: ~/Downloads
File Edit View Terminal Tabs Help

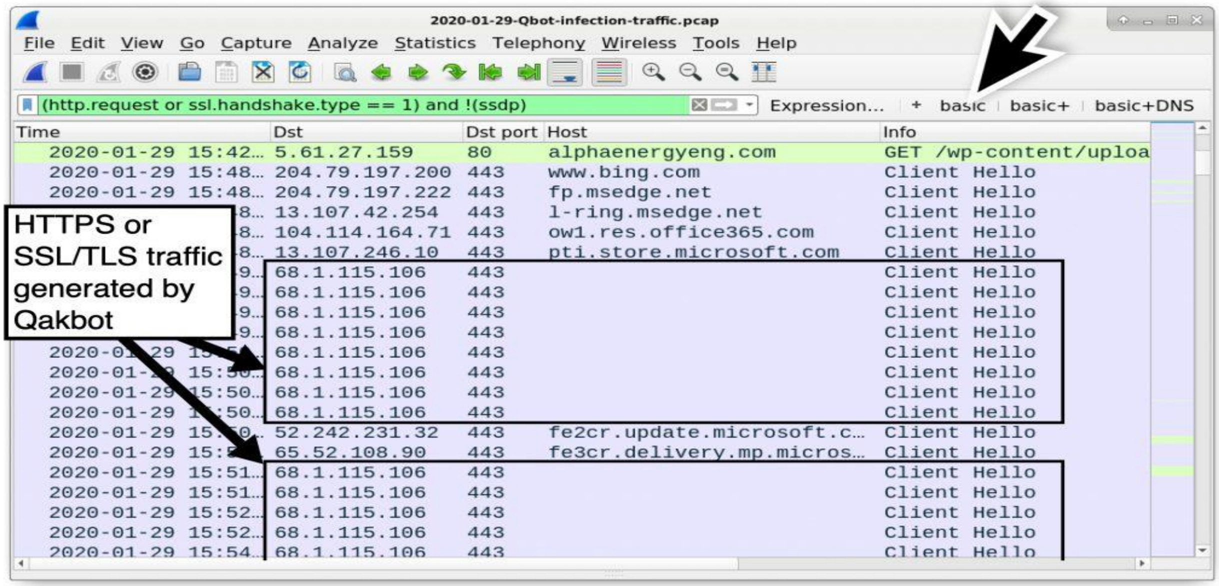
```
~/Downloads$ file 444444.png
444444.png: PE32 executable (GUI) Intel 80386, for MS Windows
~/Downloads$ shasum -a 256 444444.png
56ee803fa903ab477f939b3894af6771aebf0138abe38ae8e3c41cf96bbb0f2a  444444.png
~/Downloads$ 
```

Figure 17. Checking the exported file in a Debian-based Linux terminal window.

From our pcap, the Qakbot executable should be this file submitted to VirusTotal. A public sandbox analysis of this file generated several Qakbot indicators (identified as Qbot).

Post-infection HTTPS Activity

Use your basic filter (covered in this previous WIreshark tutorial) for a quick view of web traffic in our pcap. Scroll down to activity after the HTTP GET request to alphaenergyeng[.]com that returned our Qakbot executable. You should see several indicators of HTTPS or SSL/TLS traffic to 68.1.115[.]106 with no associated domain as noted in Figure 18.


2020-01-29-Qbot-infection-traffic.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression…
+
basic
basic+
basic+DNS
HTTPS or SSL/TLS traffic generated by Qakbot

(http.request or ssl.handshake.type == 1) and !(ssdp)

Time	Dst	Dst port	Host	Info
2020-01-29 15:42…	5.61.27.159	80	alphaenergyeng.com	GET /wp-content/uploa
2020-01-29 15:48…	204.79.197.200	443	www.bing.com	Client Hello
2020-01-29 15:48…	204.79.197.222	443	fp.msedge.net	Client Hello
8…	13.107.42.254	443	l-ring.msedge.net	Client Hello
8…	104.114.164.71	443	ow1.res.office365.com	Client Hello
8…	13.107.246.10	443	pti.store.microsoft.com	Client Hello
9…	68.1.115.106	443		Client Hello
9…	68.1.115.106	443		Client Hello
9…	68.1.115.106	443		Client Hello
9…	68.1.115.106	443		Client Hello
2020-01-29 15:5…	68.1.115.106	443		Client Hello
2020-01-29 15:50…	68.1.115.106	443		Client Hello
2020-01-29 15:50…	68.1.115.106	443		Client Hello
2020-01-29 15:50…	68.1.115.106	443		Client Hello
2020-01-29 15:50…	52.242.231.32	443	fe2cr.update.microsoft.c…	Client Hello
2020-01-29 15:5…	65.52.108.90	443	fe3cr.delivery.mp.micros…	Client Hello
2020-01-29 15:51…	68.1.115.106	443		Client Hello
2020-01-29 15:51…	68.1.115.106	443		Client Hello
2020-01-29 15:52…	68.1.115.106	443		Client Hello
2020-01-29 15:52…	68.1.115.106	443		Client Hello
2020-01-29 15:54…	68.1.115.106	443		Client Hello

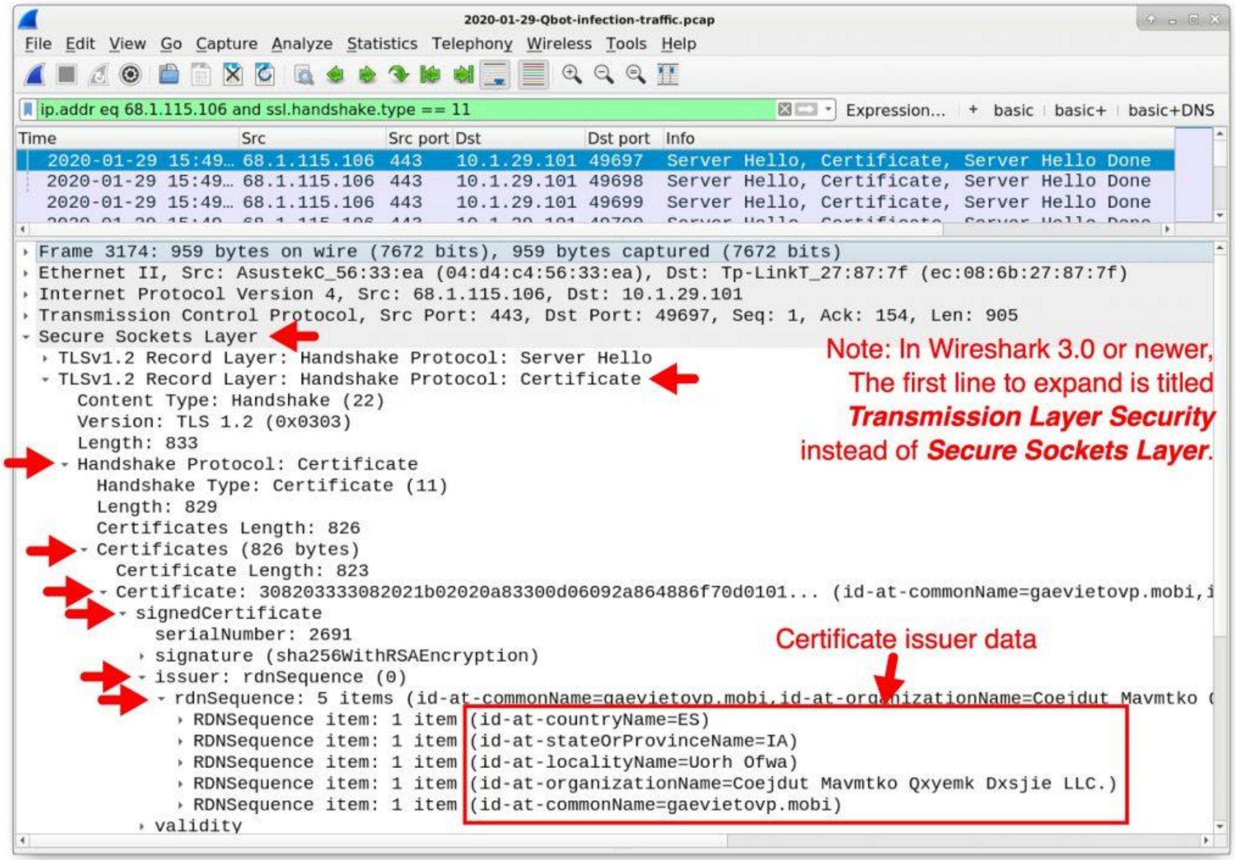
Figure 18. HTTPS or SSL/TLS traffic caused by Qakbot.

This traffic has unusual certificate issuer data commonly noted during Qakbot infections. We reviewed unusual certificate issuer data in our previous WIreshark tutorial about Ursnif, so this should be easy to find.

Let’s review our Qakbot certificate issuer data using the following Wireshark filter:

Ip.addr eq 68.1.115.186 and ssl.handshake.type eq 11

For Wireshark 3.0 or newer, use ***tls.handshake.type*** instead of ***ssl.handshake.type***. Select the first frame in your results and expand the frame details window until you find the certificate issuer data as shown in Figure 19.



```
ip.addr eq 68.1.115.106 and ssl.handshake.type == 11
```

```
Time                 Src            Src port Dst          Dst port Info
2020-01-29 15:49… 68.1.115.106 443      10.1.29.101 49697  Server Hello, Certificate, Server Hello Done
2020-01-29 15:49… 68.1.115.106 443      10.1.29.101 49698  Server Hello, Certificate, Server Hello Done
2020-01-29 15:49… 68.1.115.106 443      10.1.29.101 49699  Server Hello, Certificate, Server Hello Done
```

```
› Frame 3174: 959 bytes on wire (7672 bits), 959 bytes captured (7672 bits)
› Ethernet II, Src: AsustekC_56:33:ea (04:d4:c4:56:33:ea), Dst: Tp-LinkT_27:87:7f (ec:08:6b:27:87:7f)
› Internet Protocol Version 4, Src: 68.1.115.106, Dst: 10.1.29.101
› Transmission Control Protocol, Src Port: 443, Dst Port: 49697, Seq: 1, Ack: 154, Len: 905
▾ Secure Sockets Layer
  › TLSv1.2 Record Layer: Handshake Protocol: Server Hello
  ▾ TLSv1.2 Record Layer: Handshake Protocol: Certificate
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 833
    ▾ Handshake Protocol: Certificate
        Handshake Type: Certificate (11)
        Length: 829
        Certificates Length: 826
      ▾ Certificates (826 bytes)
          Certificate Length: 823
        ▾ Certificate: 308203333082021b02020a83300d06092a864886f70d0101... (id-at-commonName=gaevietovp.mobi,i
          ▾ signedCertificate
              serialNumber: 2691
            › signature (sha256WithRSAEncryption)
            ▾ issuer: rdnSequence (0)
              ▾ rdnSequence: 5 items (id-at-commonName=gaevietovp.mobi,id-at-organizationName=Coejdut Mavmtko (
                › RDNSequence item: 1 item (id-at-countryName=ES)
                › RDNSequence item: 1 item (id-at-stateOrProvinceName=IA)
                › RDNSequence item: 1 item (id-at-localityName=Uorh Ofwa)
                › RDNSequence item: 1 item (id-at-organizationName=Coejdut Mavmtko Qxyemk Dxsjie LLC.)
                › RDNSequence item: 1 item (id-at-commonName=gaevietovp.mobi)
            › validity
```

Note: In Wireshark 3.0 or newer, The first line to expand is titled ***Transmission Layer Security*** instead of ***Secure Sockets Layer***.

Certificate issuer data

Figure 19. Reviewing certificate issuer data from Qakbot traffic.

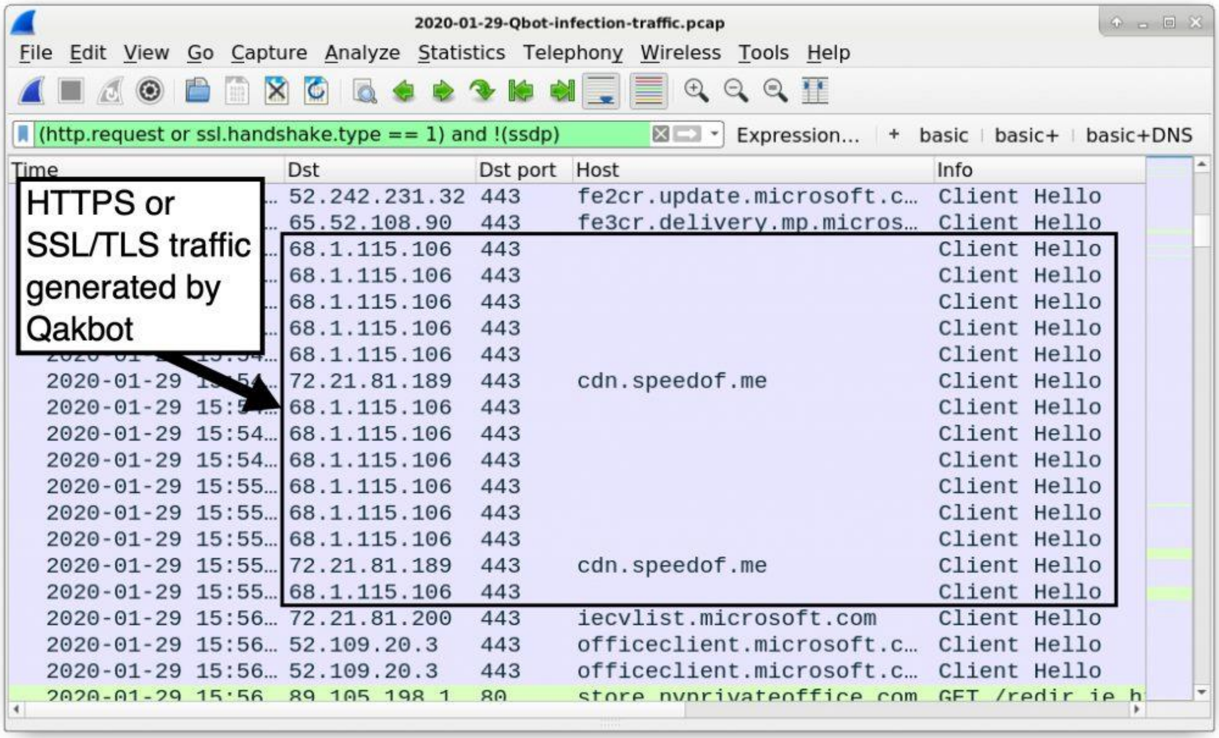
Patterns for the locality name, organization name, and common name are highly-unusual, not normally found in certificates from legitimate HTTPS, SSL, or TLS traffic. Our example of this issuer data is listed below:

- id-at-countryName=**ES**
- id-at-stateOrProvinceName=**IA**
- id-at-localityName=**Uorh Ofwa**
- id-at-organizationName=**Coejdut Mavmtko Qxyemk Dxsjie LLC.**
- id-at-commonName=**gaevietovp.mobi**

Other Post-infection Traffic

Our pcap contains other activity associated with a Qakbot infection. Each activity is not inherently malicious on its own, but taken together with our previous findings, we can assume a full Qakbot infection.

Another indicator of a Qakbot infection is HTTPS traffic to cdn.speedof[.]me. The domain speedof[.]me is used by a legitimate Internet speed test service. Although this is not malicious traffic, we frequently see traffic to cdn.speedof[.]me during Qakbot infections. Figure 20 shows this activity from our pcap.


2020-01-29-Qbot-infection-traffic.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression… + basic basic+ basic+DNS

(http.request or ssl.handshake.type == 1) and !(ssdp)

HTTPS or SSL/TLS traffic generated by Qakbot

Time	Dst	Dst port	Host	Info
…	52.242.231.32	443	fe2cr.update.microsoft.c…	Client Hello
…	65.52.108.90	443	fe3cr.delivery.mp.micros…	Client Hello
…	68.1.115.106	443		Client Hello
…	68.1.115.106	443		Client Hello
…	68.1.115.106	443		Client Hello
…	68.1.115.106	443		Client Hello
…	68.1.115.106	443		Client Hello
2020-01-29 1…	72.21.81.189	443	cdn.speedof.me	Client Hello
2020-01-29 15:…	68.1.115.106	443		Client Hello
2020-01-29 15:54…	68.1.115.106	443		Client Hello
2020-01-29 15:54…	68.1.115.106	443		Client Hello
2020-01-29 15:55…	68.1.115.106	443		Client Hello
2020-01-29 15:55…	68.1.115.106	443		Client Hello
2020-01-29 15:55…	68.1.115.106	443		Client Hello
2020-01-29 15:55…	72.21.81.189	443	cdn.speedof.me	Client Hello
2020-01-29 15:55…	68.1.115.106	443		Client Hello
2020-01-29 15:56…	72.21.81.200	443	iecvlist.microsoft.com	Client Hello
2020-01-29 15:56…	52.109.20.3	443	officeclient.microsoft.c…	Client Hello
2020-01-29 15:56…	52.109.20.3	443	officeclient.microsoft.c…	Client Hello
2020-01-29 15:56	89.105.198.1	80	store.nvprivateoffice.com	GET /redir_ie.h

Figure 20. The domain cdn.speedof[.]me within the Qakbot traffic.

Qakbot also opens windows from all browsers on an infected Windows host. At approximately 13 minutes and 5 seconds into this sandbox analysis, the video playback shows Qakbot opening Chrome, then Firefox, then Internet Explorer on a Windows 7 host. This analysis shows Qakbot generated traffic to the following URLs:

- hxxp://store.nvprivateoffice[.]com/redir_chrome.html
- hxxp://store.nvprivateoffice[.]com/redir_ff.html
- hxxp://store.nvprivateoffice[.]com/redir_ie.html

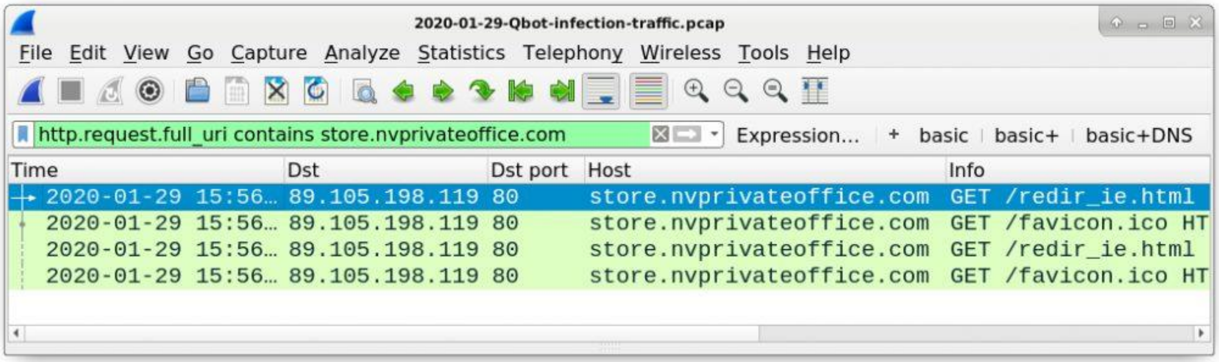
The domain nvprivateoffice[.]com has been registered through GoDaddy since 2012, and store.nvprivateoffice[.]com shows a default web page for nginx on a Fedora server.

Our pcap for this tutorial is from a Qakbot infection on a Windows 10 host without Chrome or Firefox installed. Our pcap only shows web traffic for Internet Explorer and the new Chromium-based Microsoft Edge. Both times,

the URL generated by Qakbot
was hxxp://store.nvprivateoffice[.]com/redir_ie.html.

To find this traffic, use the following Wireshark filter as shown in Figure 21:

http.request.full_uri contains store.nvprivateoffice

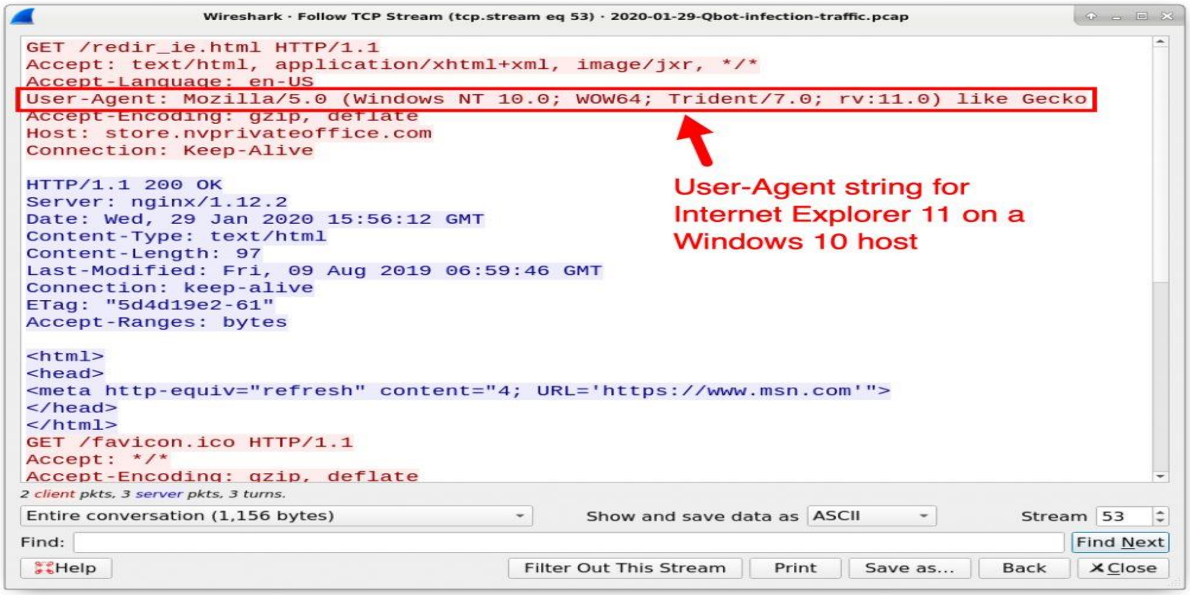

2020-01-29-Qbot-infection-traffic.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression… + basic basic+ basic+DNS

http.request.full_uri contains store.nvprivateoffice.com

Time	Dst	Dst port	Host	Info
2020-01-29 15:56…	89.105.198.119	80	store.nvprivateoffice.com	GET /redir_ie.html
2020-01-29 15:56…	89.105.198.119	80	store.nvprivateoffice.com	GET /favicon.ico HT
2020-01-29 15:56…	89.105.198.119	80	store.nvprivateoffice.com	GET /redir_ie.html
2020-01-29 15:56…	89.105.198.119	80	store.nvprivateoffice.com	GET /favicon.ico HT

Figure 21. Finding Qakbot traffic that opens web browsers on an infected Windows host.

Follow the TCP stream for each of the two HTTP GET requests ending in redir_ie.html. The first request has a User-Agent in the HTTP headers for Internet Explorer as shown in Figure 22. The second request for the same URL has a User-Agent in the HTTP headers for the new Chromium-based Microsoft Edge as noted in Figure 23.


Wireshark · Follow TCP Stream (tcp.stream eq 53) · 2020-01-29-Qbot-infection-traffic.pcap

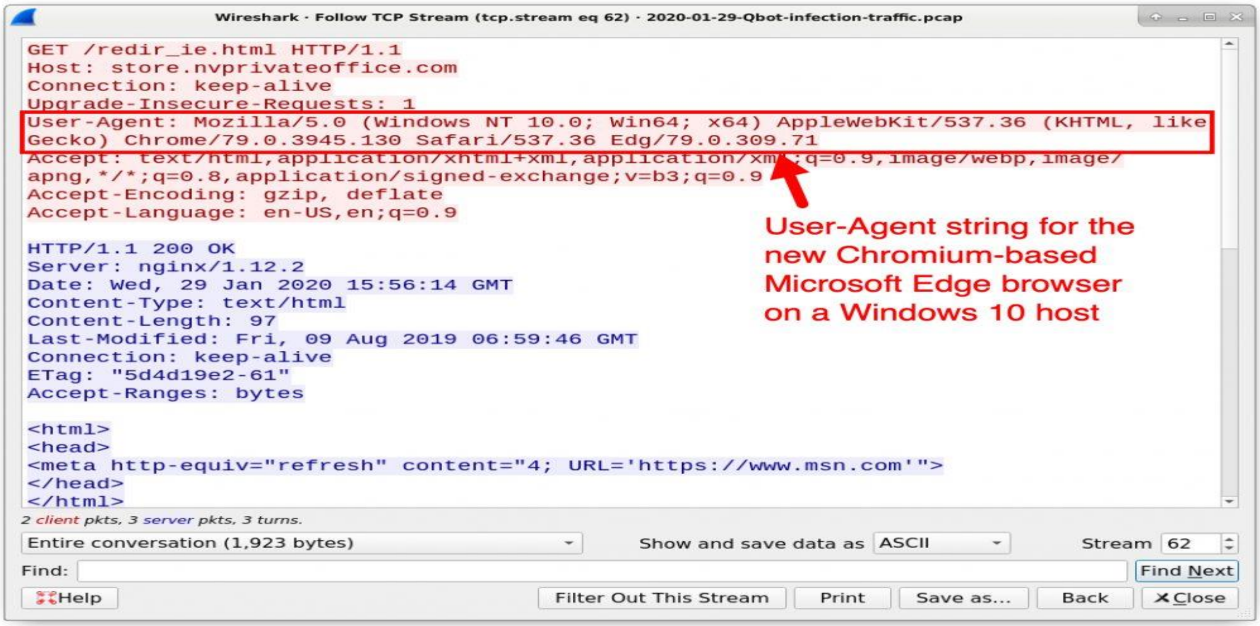
```
GET /redir_ie.html HTTP/1.1
Accept: text/html, application/xhtml+xml, image/jxr, */*
Accept-Language: en-US
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko
Accept-Encoding: gzip, deflate
Host: store.nvprivateoffice.com
Connection: Keep-Alive

HTTP/1.1 200 OK
Server: nginx/1.12.2
Date: Wed, 29 Jan 2020 15:56:12 GMT
Content-Type: text/html
Content-Length: 97
Last-Modified: Fri, 09 Aug 2019 06:59:46 GMT
Connection: keep-alive
ETag: "5d4d19e2-61"
Accept-Ranges: bytes

<html>
<head>
<meta http-equiv="refresh" content="4; URL='https://www.msn.com'">
</head>
</html>
GET /favicon.ico HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
```

User-Agent string for Internet Explorer 11 on a Windows 10 host

2 client pkts, 3 server pkts, 3 turns.

Entire conversation (1,156 bytes) Show and save data as ASCII Stream 53

Figure 22. Qakbot traffic to store.nvprivateoffice[.]com using Internet Explorer 11.



Wireshark · Follow TCP Stream (tcp.stream eq 62) · 2020-01-29-Qbot-infection-traffic.pcap

```
GET /redir_ie.html HTTP/1.1
Host: store.nvprivateoffice.com
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/79.0.3945.130 Safari/537.36 Edg/79.0.309.71
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/
apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

HTTP/1.1 200 OK
Server: nginx/1.12.2
Date: Wed, 29 Jan 2020 15:56:14 GMT
Content-Type: text/html
Content-Length: 97
Last-Modified: Fri, 09 Aug 2019 06:59:46 GMT
Connection: keep-alive
ETag: "5d4d19e2-61"
Accept-Ranges: bytes

<html>
<head>
<meta http-equiv="refresh" content="4; URL='https://www.msn.com'">
</head>
</html>
```

User-Agent string for the new Chromium-based Microsoft Edge browser on a Windows 10 host

2 client pkts, 3 server pkts, 3 turns.

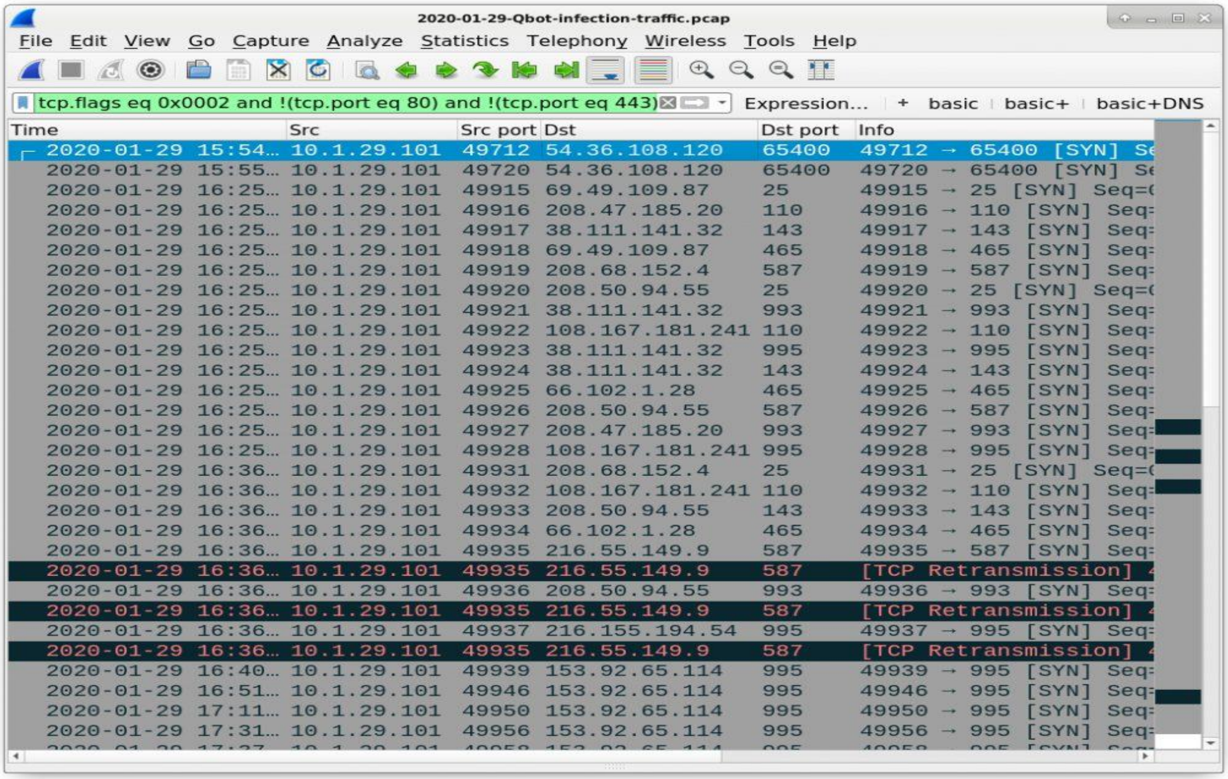
Entire conversation (1,923 bytes) · Show and save data as: ASCII · Stream: 62

Find:

Figure 23. Qakbot traffic to store.nvprivateoffice[.]com using the new Chromium-based Microsoft Edge.

Finally, our pcap from the Qakbot-infected host also has email-related TCP traffic to various ports for various email protocols like SMTP, IMAP, and POP3. To get an idea of this non-web-related traffic, use the following Wireshark filter as shown in Figure 25:

tcp.flags eq 0x0002 and !(tcp.port eq 80) and !(tcp.port eq 443)


2020-01-29-Qbot-infection-traffic.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression…
+
basic
basic+
basic+DNS

tcp.flags eq 0x0002 and !(tcp.port eq 80) and !(tcp.port eq 443)

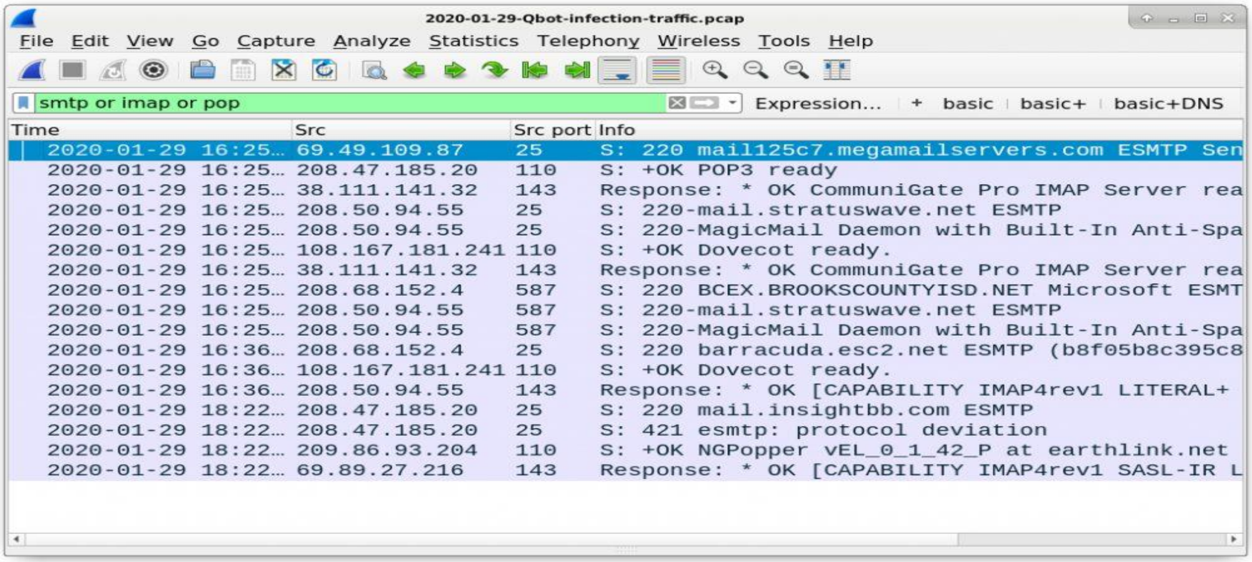
Time	Src	Src port	Dst	Dst port	Info
2020-01-29 15:54…	10.1.29.101	49712	54.36.108.120	65400	49712 → 65400 [SYN] Se
2020-01-29 15:55…	10.1.29.101	49720	54.36.108.120	65400	49720 → 65400 [SYN] Se
2020-01-29 16:25…	10.1.29.101	49915	69.49.109.87	25	49915 → 25 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49916	208.47.185.20	110	49916 → 110 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49917	38.111.141.32	143	49917 → 143 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49918	69.49.109.87	465	49918 → 465 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49919	208.68.152.4	587	49919 → 587 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49920	208.50.94.55	25	49920 → 25 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49921	38.111.141.32	993	49921 → 993 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49922	108.167.181.241	110	49922 → 110 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49923	38.111.141.32	995	49923 → 995 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49924	38.111.141.32	143	49924 → 143 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49925	66.102.1.28	465	49925 → 465 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49926	208.50.94.55	587	49926 → 587 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49927	208.47.185.20	993	49927 → 993 [SYN] Seq=
2020-01-29 16:25…	10.1.29.101	49928	108.167.181.241	995	49928 → 995 [SYN] Seq=
2020-01-29 16:36…	10.1.29.101	49931	208.68.152.4	25	49931 → 25 [SYN] Seq=
2020-01-29 16:36…	10.1.29.101	49932	108.167.181.241	110	49932 → 110 [SYN] Seq=
2020-01-29 16:36…	10.1.29.101	49933	208.50.94.55	143	49933 → 143 [SYN] Seq=
2020-01-29 16:36…	10.1.29.101	49934	66.102.1.28	465	49934 → 465 [SYN] Seq=
2020-01-29 16:36…	10.1.29.101	49935	216.55.149.9	587	49935 → 587 [SYN] Seq=
2020-01-29 16:36…	10.1.29.101	49935	216.55.149.9	587	[TCP Retransmission] 4
2020-01-29 16:36…	10.1.29.101	49936	208.50.94.55	993	49936 → 993 [SYN] Seq=
2020-01-29 16:36…	10.1.29.101	49935	216.55.149.9	587	[TCP Retransmission] 4
2020-01-29 16:36…	10.1.29.101	49937	216.155.194.54	995	49937 → 995 [SYN] Seq=
2020-01-29 16:36…	10.1.29.101	49935	216.55.149.9	587	[TCP Retransmission] 4
2020-01-29 16:40…	10.1.29.101	49939	153.92.65.114	995	49939 → 995 [SYN] Seq=
2020-01-29 16:51…	10.1.29.101	49946	153.92.65.114	995	49946 → 995 [SYN] Seq=
2020-01-29 17:11…	10.1.29.101	49950	153.92.65.114	995	49950 → 995 [SYN] Seq=
2020-01-29 17:31…	10.1.29.101	49956	153.92.65.114	995	49956 → 995 [SYN] Seq=

Figure 25. Getting an idea of the non-web-related traffic from this Qakbot infection.

Figure 25 shows TCP connections and attempted TCP connections to various ports like 25, 110,143, 465, 587, 993, and 995 commonly used by different email protocols. The first two lines in the results show traffic to TCP port 65400, but reviewing the associated TCP streams indicates this also email-related traffic.

Use the following Wireshark filter to get a better idea of email-related traffic from the infected host as shown in Figure 26:

smtp or imap or pop


2020-01-29-Qbot-infection-traffic.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression… + basic basic+ basic+DNS

smtp or imap or pop

```
Time                    Src             Src port Info
2020-01-29 16:25… 69.49.109.87     25     S: 220 mail125c7.megamailservers.com ESMTP Sen
2020-01-29 16:25… 208.47.185.20    110    S: +OK POP3 ready
2020-01-29 16:25… 38.111.141.32    143    Response: * OK CommuniGate Pro IMAP Server rea
2020-01-29 16:25… 208.50.94.55     25     S: 220-mail.stratuswave.net ESMTP
2020-01-29 16:25… 208.50.94.55     25     S: 220-MagicMail Daemon with Built-In Anti-Spa
2020-01-29 16:25… 108.167.181.241 110    S: +OK Dovecot ready.
2020-01-29 16:25… 38.111.141.32    143    Response: * OK CommuniGate Pro IMAP Server rea
2020-01-29 16:25… 208.68.152.4     587    S: 220 BCEX.BROOKSCOUNTYISD.NET Microsoft ESMT
2020-01-29 16:25… 208.50.94.55     587    S: 220-mail.stratuswave.net ESMTP
2020-01-29 16:25… 208.50.94.55     587    S: 220-MagicMail Daemon with Built-In Anti-Spa
2020-01-29 16:36… 208.68.152.4     25     S: 220 barracuda.esc2.net ESMTP (b8f05b8c395c8
2020-01-29 16:36… 108.167.181.241 110    S: +OK Dovecot ready.
2020-01-29 16:36… 208.50.94.55     143    Response: * OK [CAPABILITY IMAP4rev1 LITERAL+
2020-01-29 18:22… 208.47.185.20    25     S: 220 mail.insightbb.com ESMTP
2020-01-29 18:22… 208.47.185.20    25     S: 421 esmtp: protocol deviation
2020-01-29 18:22… 209.86.93.204    110    S: +OK NGPopper vEL_0_1_42_P at earthlink.net
2020-01-29 18:22… 69.89.27.216     143    Response: * OK [CAPABILITY IMAP4rev1 SASL-IR L
```

Figure 26. Finding email-related traffic caused by Qakbot in this pcap.

Follow some of the TCP streams to get a better idea for this type of email traffic. We do not normally see such unencrypted email traffic originating from a Windows client to public IP addresses. Along with other indicators, this ***smtp or imap or pop*** filter may reveal Qakbot activity.