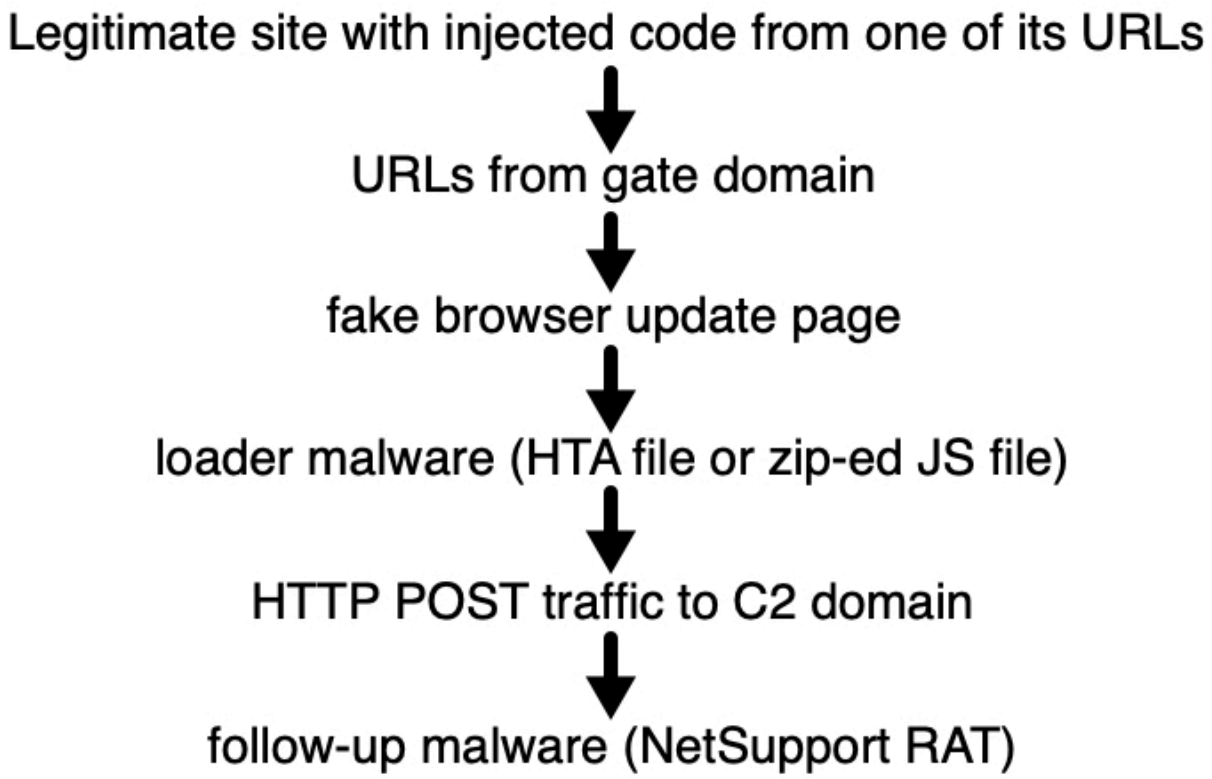
Fake browser update pages are

By SANS Internet Storm Center

Archived: 2026-04-05 19:37:02 UTC

Introduction

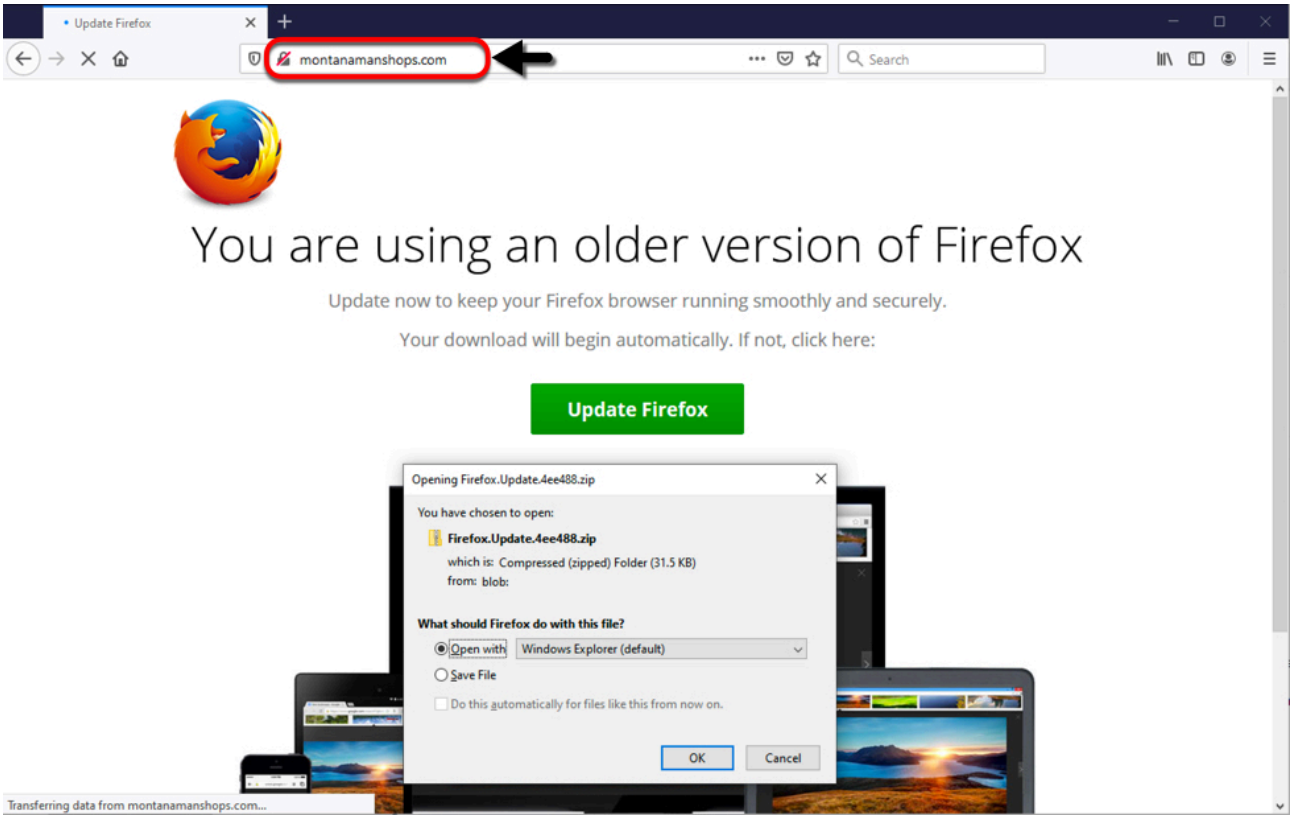
SocGholish is a term I first saw in signatures from the EmergingThreats Pro ruleset to describe fake browser update pages used to distribute malware like a NetSupport RAT-based malware package or Chthonic banking malware. Although this activity has continued into 2020, I hadn't run across an example until this week.



Shown above: A recent infection chain from the SocGholish campaign.

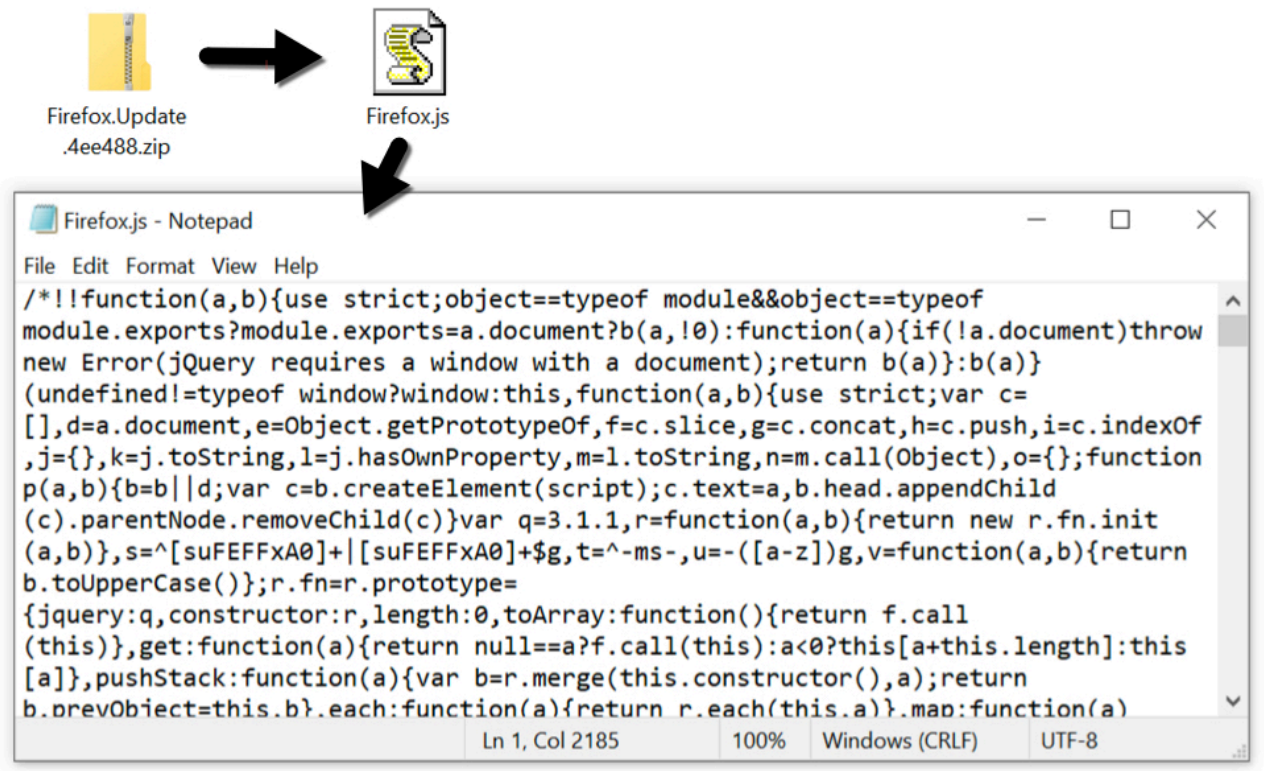
Fake browser update pages

The beginning of an infection chain starts with a legitimate website with injected code from a file sent by of its URLs. The URL most often ends with a ***.js***. The injected code is highly-obfuscated, and I was unable to figure out where it came from on the legitimate site when I generated an infection in my lab. The end result looked like the image below.



Shown above: Fake browser update page seen after visiting a legitimate website.

The downloaded zip archive contained a JavaScript file with heavily obfuscated Javascript. This happened when I used Firefox as my web browser. If you use Google Chrome, the fake browser page sends an HTA file instead of a zip archive. In my example, the fake Firefox update page sent a zip archive containing a file named ***Firefox.js*** for the malware downloader.



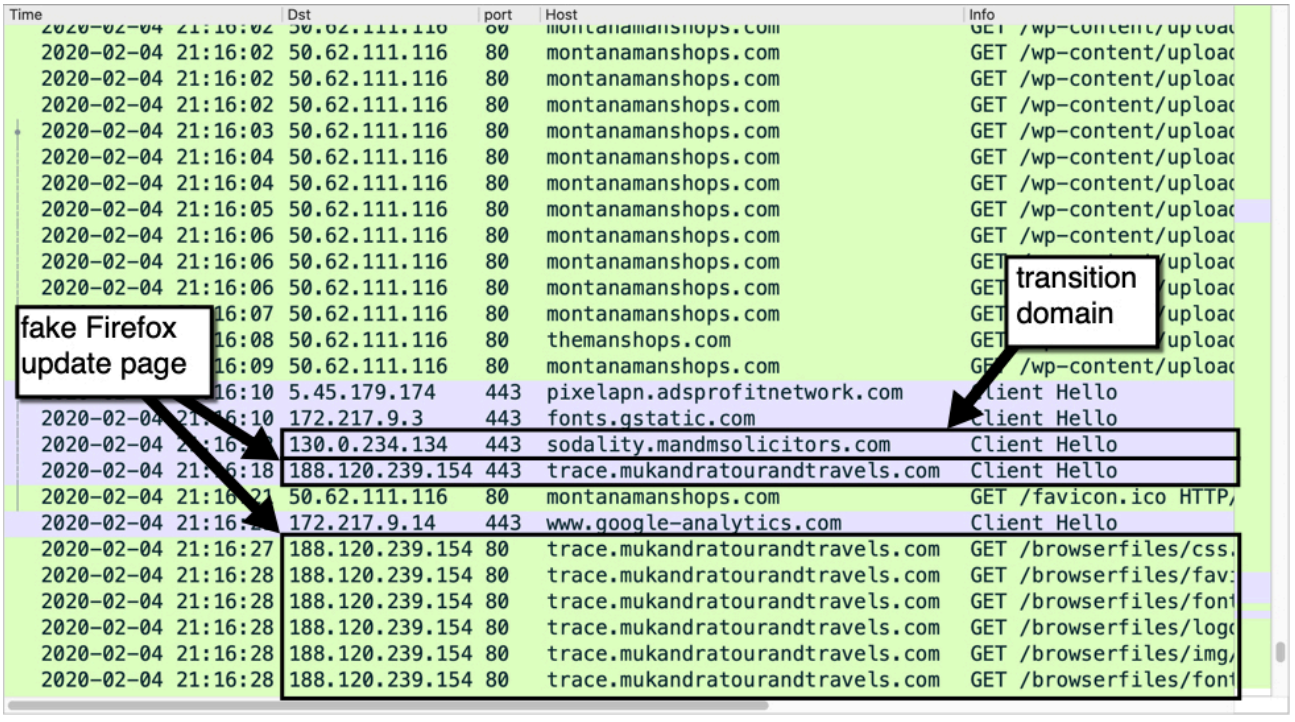
Shown above: The downloaded zip archive and extracted .js file.

Infection traffic

Infection traffic was typical of what I've seen before with this campaign. The malware downloader is *very* picky. It knows which machines I've infected before, so when I use a computer that I've infected once or twice before, it won't deliver the follow-up malware. Also, this .js-based downloader (or HTA-based downloader if you had a fake Chrome update page) is extremely VM-aware. It's rare for me to get a full infection chain of events. In this case, I got the fake browser update page on one computer, then I switched to another computer to get ***Firefox.js*** to deliver the follow-up malware.

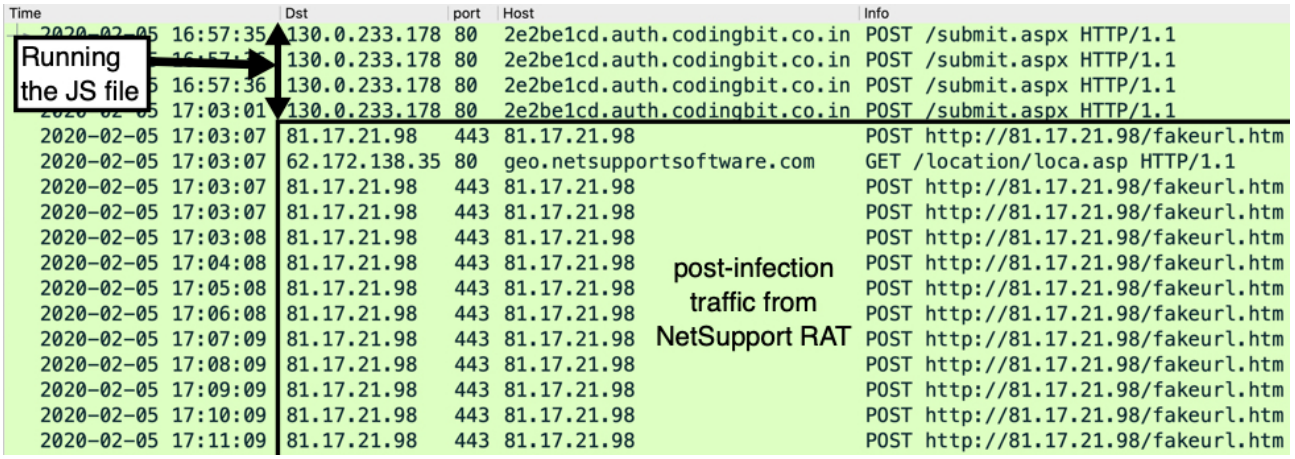
#	Result	Protocol	Host	URL	Body	Content-Type
47	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/Missoula-Inside-150x140....	19,580	image/jpeg
48	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/Missoula-Inside-2-150x14...	18,750	image/jpeg
49	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/08/one.png	30,830	image/png
50	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/Missoula-Inside-3-150x15...	20,535	image/jpeg
51	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/s1.png	34,084	image/png
52	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/s4-150x149.png	0	image/png
53	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/s3-150x149.png	39,587	image/png
54	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/s2.png	31,984	image/png
55	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/p44.png	33,	
56	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/p33.png	39,	
57	200	HTTP	montanamanshops.com	/wp-content/uploads/2017/09/p22.png	31,	
58	200	HTTP	montanamanshops.com	/wp-content/uploads/2018/10/Slide4.png	7,	
			montanamanshops.com	/wp-content/uploads/2017/09/p11.png		image/png
			pixelapn.adsprofitnetwork.com	/apnpixel.png?ti=8sw=1440&sh=900&c=1358&cd=24&...	258	image/png
			fonts.gstatic.com	/s/opensans/v17/mem8YaGs126MiZpBA-UFVZ0b.woff2	14,380	font/woff2
			sodality.mandmsolicitors.com	/WebResource.axd?d=dj1iODNkMDU1YzUzZWRhZDkyNj...	2,245	application/java
63	200	HTTPS	sodality.mandmsolicitors.com	/WebResource.axd?d=Y2lkPTI0NyZ2PTgyNTAxNzYyYTE...	3,919	application/java
64	200	HTTPS	sodality.mandmsolicitors.com	/WebResource.axd?d=Y2lkPTI0NyZ2PWQ5YzcwNWU2Z...	2,008	application/java
65	200	HTTPS	trace.mukandratourandtravels.com	/wordpress/index.php?a=247&c=377366&q=4b834bb8...	47,097	text/html; chars
66	200	HTTPS	montanamanshops.com	/favicon.ico	0	image/vnd.micro
67	200	HTTPS	www.google-analytics.com	/ga.js	17,168	text/javascript
68	200	HTTPS	www.google-analytics.com	/r/__utm.gif?utmwv=5.7.2&utms=1&utmn=374209818...	35	image/gif
69	200	HTTP	trace.mukandratourandtravels.com	/browserfiles/css.css	12,870	text/css
70	200	HTTP	trace.mukandratourandtravels.com	/browserfiles/favicon/firefox.ico	5,430	image/x-icon
71	200	HTTP	trace.mukandratourandtravels.com	/browserfiles/logo/firefox.png	6,069	image/png
72	200	HTTP	trace.mukandratourandtravels.com	/browserfiles/img/chrome.jpg	68,716	image/jpeg
73	200	HTTP	trace.mukandratourandtravels.com	/browserfiles/fonts/cJZKeOuBrn4kERxqtaUH3VtXRa8TV...	15,572	font/woff2
74	200	HTTP	trace.mukandratourandtravels.com	/browserfiles/fonts/MTP_ySUJH_bn48VBG8sNSugdm0LZ...	16,164	font/woff2
75	200	HTTP	trace.mukandratourandtravels.com	/browserfiles/fonts/DXI1ORHCpsQm3Vp6mXoaTegdm0L...	16,152	font/woff2
76	200	HTTP	trace.mukandratourandtravels.com	/browserfiles/fonts/k3k702ZOKiLJc3WVjuplzOgdm0LZdjq...	16,276	font/woff2
77	200	HTTP	trace.mukandratourandtravels.com	/wordpress/index.php?a=247&c=377366&q=4b834bb8...	35	image/gif

Shown above: Gate URLs and a fake Firefox update page from the SocGholish campaign shown in a [Fiddler](#) capture.

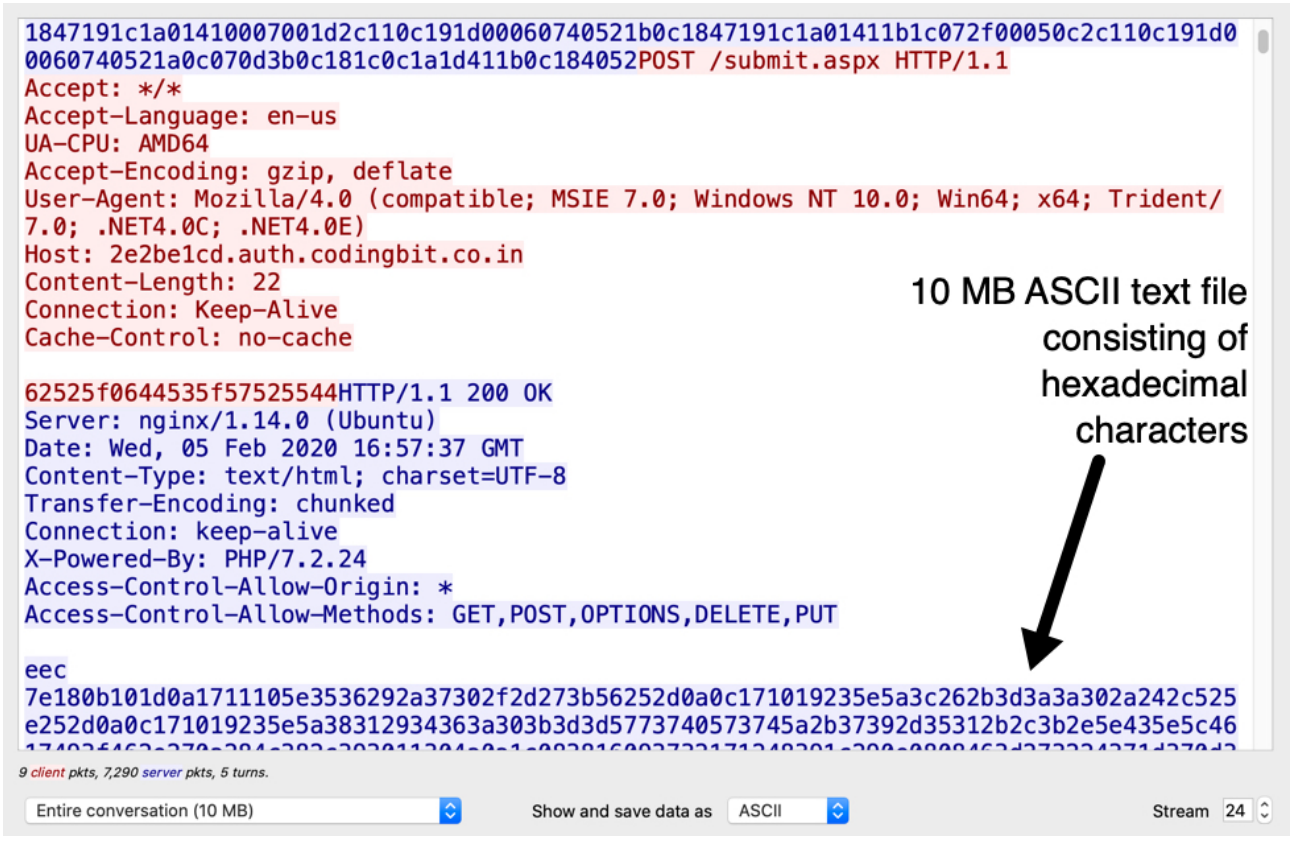


Time	Dst	port	Host	Info
2020-02-04 21:16:02	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:02	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:02	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:03	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:04	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:04	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:05	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:06	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:06	50.62.111.116	80	montanamanshops.com	GET /uploac
2020-02-04 21:16:06	50.62.111.116	80	montanamanshops.com	GET /uploac
2020-02-04 21:16:07	50.62.111.116	80	montanamanshops.com	GET /uploac
2020-02-04 21:16:08	50.62.111.116	80	themanshops.com	GET /uploac
2020-02-04 21:16:09	50.62.111.116	80	montanamanshops.com	GET /wp-content/uploac
2020-02-04 21:16:10	5.45.179.174	443	pixelapn.adsprofitnetwork.com	Client Hello
2020-02-04 21:16:10	172.217.9.3	443	fonts.gstatic.com	Client Hello
2020-02-04 21:16:17	130.0.234.134	443	sodality.mandmsolicitors.com	Client Hello
2020-02-04 21:16:18	188.120.239.154	443	trace.mukandratourandtravels.com	Client Hello
2020-02-04 21:16:21	50.62.111.116	80	montanamanshops.com	GET /favicon.ico HTTP/
2020-02-04 21:16:21	172.217.9.14	443	www.google-analytics.com	Client Hello
2020-02-04 21:16:27	188.120.239.154	80	trace.mukandratourandtravels.com	GET /browserfiles/css.
2020-02-04 21:16:28	188.120.239.154	80	trace.mukandratourandtravels.com	GET /browserfiles/fav
2020-02-04 21:16:28	188.120.239.154	80	trace.mukandratourandtravels.com	GET /browserfiles/font
2020-02-04 21:16:28	188.120.239.154	80	trace.mukandratourandtravels.com	GET /browserfiles/logo
2020-02-04 21:16:28	188.120.239.154	80	trace.mukandratourandtravels.com	GET /browserfiles/img/
2020-02-04 21:16:28	188.120.239.154	80	trace.mukandratourandtravels.com	GET /browserfiles/font

Shown above: Gate domain and fake Firefox update page from the SocGholish campaign from a pcap shown in Wireshark.



*Shown above: Traffic from infecting a host with the **Firefox.js** file.*

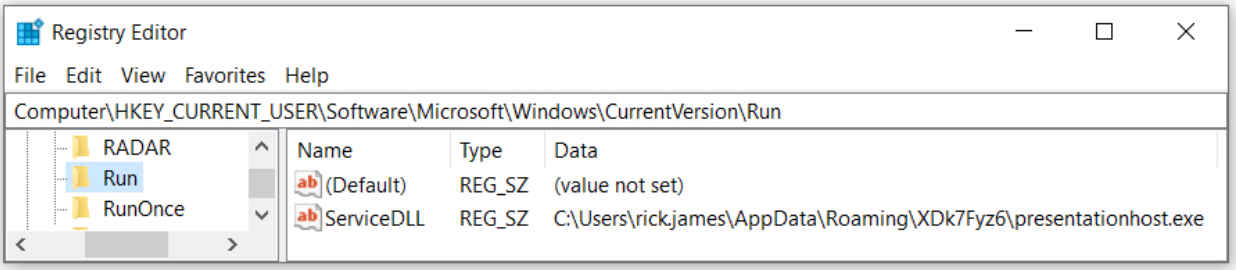


*Shown above: Data returned from the server contacted after running **Firefox.js** on my lab host.*

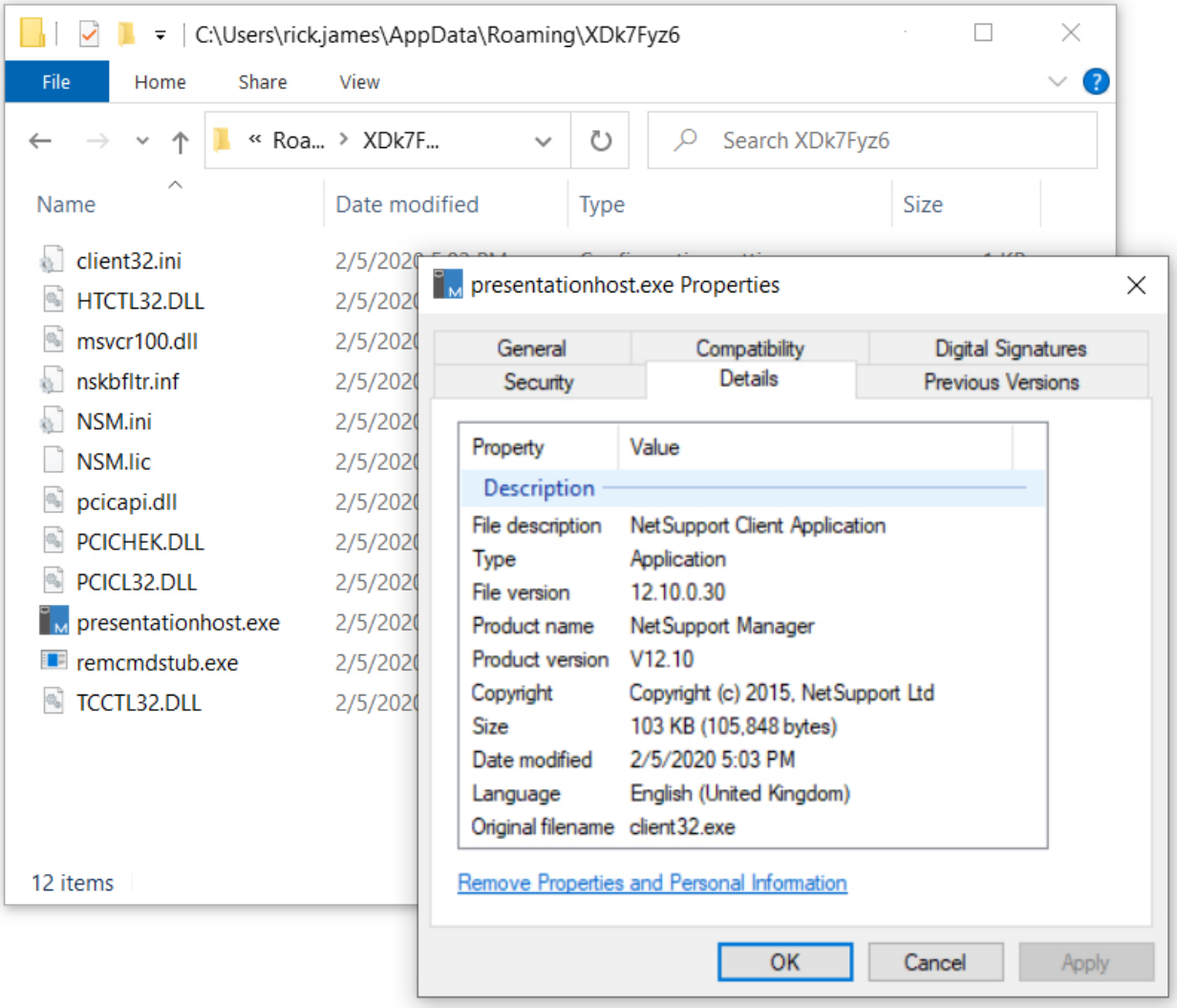
This NetSupport RAT-based malware package was sent as a 10MB ASCII text file consisting of hexadecimal characters. This is encoded data, and the file was saved to my lab host and decoded to a zip archive containing the malware package. This ASCII data and decoded zip archive were deleted from my infected lab host by the time I performed post-infection forensics.

Post-infection forensics

The NetSupport RAT-based malware package was kept persistent through the Windows registry and stored in a folder under the infected user's ***AppData\Roaming*** directory.



Shown above: NetSupport RAT-based malware package persistent on the infected windows host.



Shown above: NetSupport RAT-based malware package stored under the infected user's AppData\Roaming directory.

Indicators from the infection

Gate activity leading to fake browser update page:

- 130.0.234[.]134 port 443 - ***sodality.mandmsolicitors[.]com*** - URLs from gate domain (HTTPS)

Fake browser update page:

- 188.120.239[.]154 port 443 - ***trace.mukandratourandtravels[.]com*** - initial URL sent as HTTPS
- 188.120.239[.]154 port 80 - ***trace.mukandratourandtravels[.]com*** - follow-up URLs for fake browser update page
- Note: The domain name used for these fake update pages frequently changes.

URLs caused by Firefox.js (malware downloader):

- 130.0.233[.]178 port 80 - ***2e2be1cd.auth.codingbit[.]co[.]in*** - POST /submit.aspx
- Note: The first part of the domain name (with the hex characters) is different for each infection.

Traffic generated by NetSupport RAT-based malware package:

- 81.17.21[.]98 port 443 - ***81.17.21[.]98*** - POST http://81.17.21[.]98/fakeurl.htm
- 62.172.138[.]35 port 80 - ***geo.netsupportsoftware[.]com*** - GET /location/loca.asp (not inherently malicious)

SHA256 hash: 6b89a2c1650012d7953f04f39ef7ecd97341114480918602d041593a597442d7

- File size: 32,231 bytes
- File name: Firefox.Update.4ee488.zip
- File description: Zip archive sent by fake browser update page
- Note: File name is different for each download (file hash might be as well)

SHA256 hash: 69ea88be502bd00e87aef75e1f41da3e5e0bdb6946d18db5a4a52d919e2dc79b

- File size: 90,690 bytes
- File name: Firefox.js
- File description: JavaScript-based malware downloader extracted from downloaded zip archive
- Note: File hash might be different on each occasion

SHA256 hash: 49a568f8ac11173e3a0d76cff6bc1d4b9bdf2c35c6d8570177422f142dcfdbe3

- File size: 105,848 bytes
- File location: C:\Users*[username]*\AppData\Roaming\XDk7Fyz6\presentationhost.exe
- File description: NetSupport Manager RAT executable

Final words

This is a long-running campaign that continually evolves. To get an idea how it has changed since last year, view my previous ISC diary I wrote about this campaign in February 2019.

Computers running Windows 10 with the latest updates and recommended security settings are not very vulnerable to this threat. Default security settings for Chrome and Firefox usually block this activity. However, the criminals behind this campaign keep updating their tactics as they attempt to evade detection, and these fake browser pages sometimes slip through. If someone clicked through enough security warnings, they might very well infect a vulnerable Windows host.

The associated malware and a pcap of the traffic can be found here.

Brad Duncan
brad [at] malware-traffic-analysis.net
