

TryHackMe: Carnage

Write-Up

Pavel Pecheniuk

1. Introduction

In the Carnage challenge by THM, one is offered to apply the analytical skills to analyse the malicious network traffic using Wireshark, a well-known tool used for network traffic analysis.

2. Scenario

Eric Fischer from the Purchasing Department at Bartell Ltd has received an email from a known contact with a Word document attachment. Upon opening the document, he accidentally clicked on "Enable Content." The SOC Department immediately received an alert from the endpoint agent that Eric's workstation was making suspicious connections outbound. The pcap was retrieved from the network sensor and handed to me for analysis.

Task: Investigate the packet capture and uncover the malicious activities.

*Credit goes to Brad Duncan for capturing the traffic and sharing the pcap packet capture with InfoSec community.

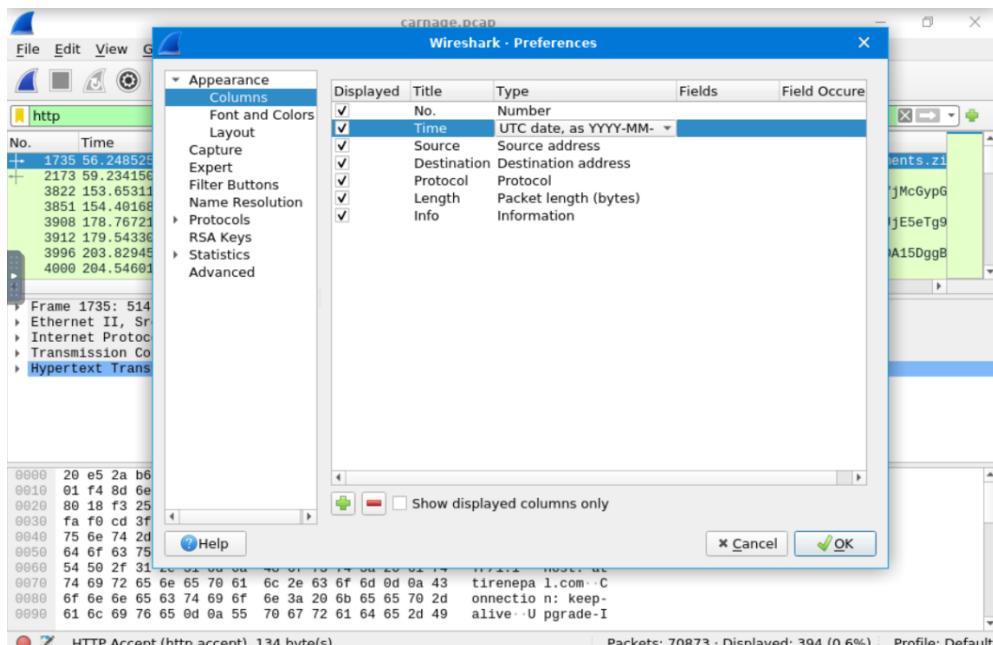
NOTE: DO NOT directly interact with any domains and IP addresses in this challenge.

3. Traffic Analysis

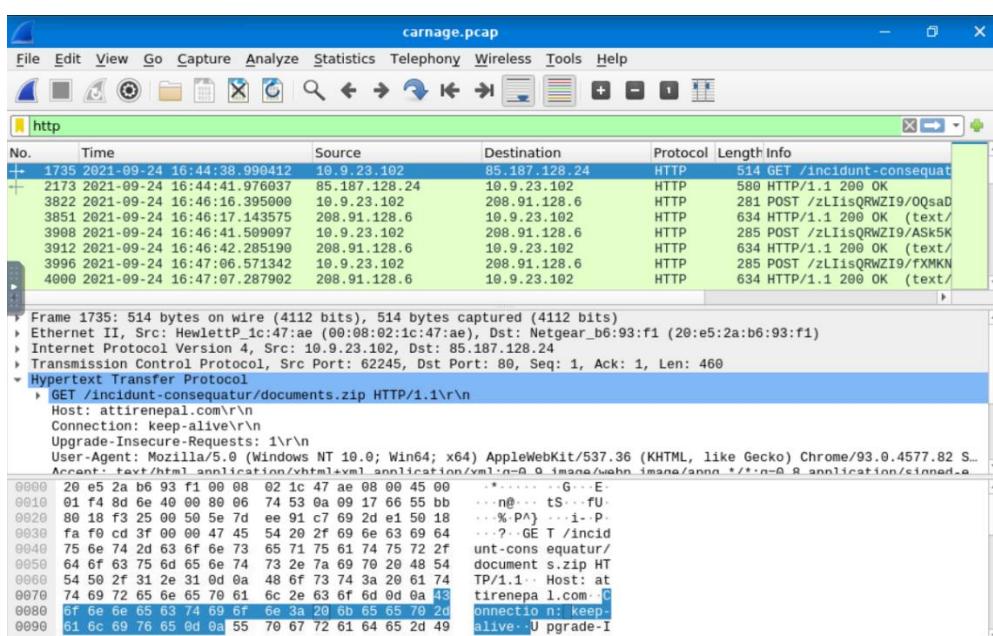
What was the date and time for the first HTTP connection to the malicious IP? (answer format: yyyy-mm-dd hh:mm:ss)

Firstly, HTTP traffic is a traffic of interest, so let's set the filter "http" to print out only HTTP packets.

Then one can observe inappropriate timing format. Configuration of the right time format is done via *Edit > Preferences > Columns > Time > UTC date, as YYYY-MM-DD, and time > OK*.



Finally, the needed result is obtained.



Answer: 2021-09-24 16:44:38

What is the name of the zip file that was downloaded?

Answer to this question can be found on the second screenshot from the previous task. One can analyse the first HTTP packet and obtain the name of a zip file.

Answer: documents.zip

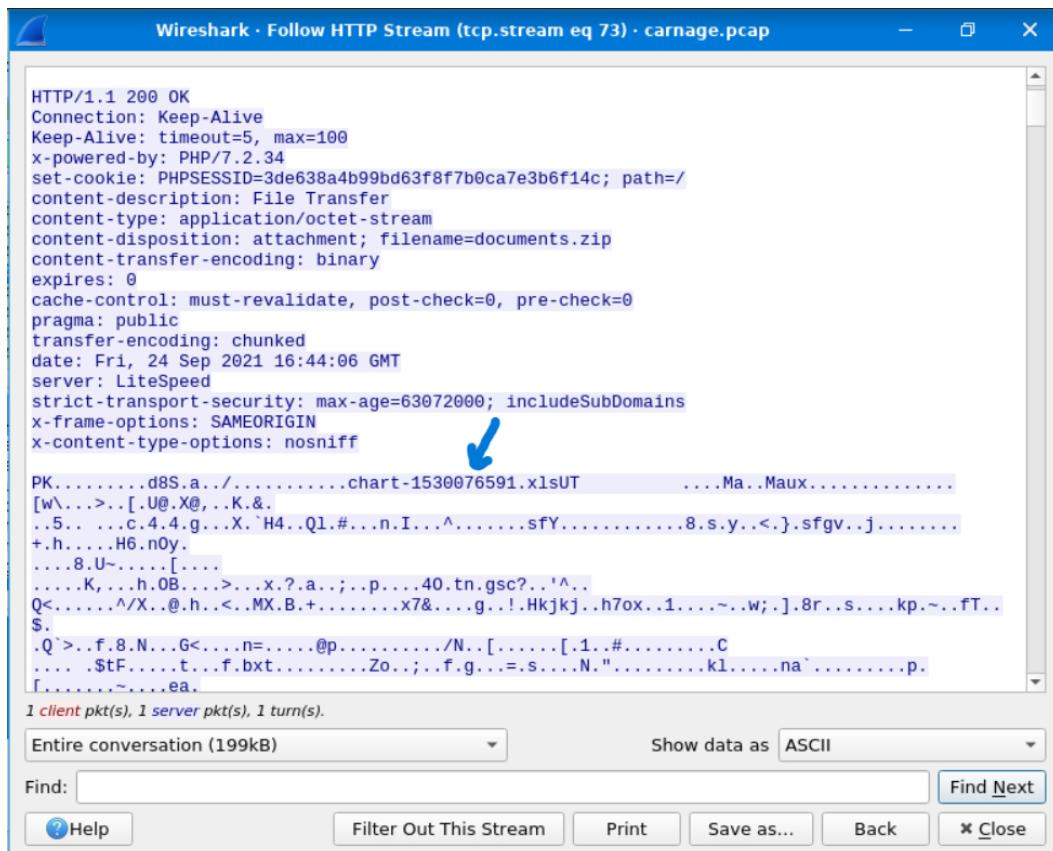
What was the domain hosting the malicious zip file?

Answer can be found again on the second screenshot from the first task. The name of the malicious domain is located under the field “Host”.

Answer: attirenepal.com

Without downloading the file, what is the name of the file in the zip file?

Let's investigate this HTTP frame to gain the needed information by right-clicking the frame and select *Follow > Follow HTTP Stream*. Here is the result containing more useful information, including the needed name.



HTTP/1.1 200 OK
Connection: Keep-Alive
Keep-Alive: timeout=5, max=100
x-powered-by: PHP/7.2.34
set-cookie: PHPSESSID=3de638a4b99bd63f8f7b0ca7e3b6f14c; path=/
content-description: File Transfer
content-type: application/octet-stream
content-disposition: attachment; filename=documents.zip
content-transfer-encoding: binary
expires: 0
cache-control: must-revalidate, post-check=0, pre-check=0
pragma: public
transfer-encoding: chunked
date: Fri, 24 Sep 2021 16:44:06 GMT
server: LiteSpeed
strict-transport-security: max-age=63072000; includeSubDomains
x-frame-options: SAMEORIGIN
x-content-type-options: nosniff

PK.....d8S.a./.....chart-1530076591.xlsUT ..Ma..Maux.....
[w\...>.[U@.X@..K.&.
.5....c.4.4.g..X.^H4..Q1.#..n.I...^.....sfY.....8.s.y.<}.sfgv..j.....
+h....H6.noy.
....8.U~....[....
....K,...h.0B....>....x.?a...;...p....40.tn.gsc?..'^..
Q<....'^/X..@.h..<..MX.B.+....x7&....g..!.Hkjkj..h7ox..1....~..w;..].8r..s....kp.~..fT..
S.
.Q`>..f.8.N...G<....n=....@p...../N..[.....[.1..#.....C
....\$.F.....t..f.bxt.....Zo...;..f.g...=s....N.".....kl....na`.....p.
.....~....ea.

1 client pkt(s), 1 server pkt(s), 1 turn(s).

Entire conversation (199kB) Show data as ASCII

Find: Find Next

? Help Filter Out This Stream Print Save as... Back Close

Answer: chart-1530076591.xls

What is the name of the webserver of the malicious IP from which the zip file was downloaded? What is the version of the webserver from the previous question?

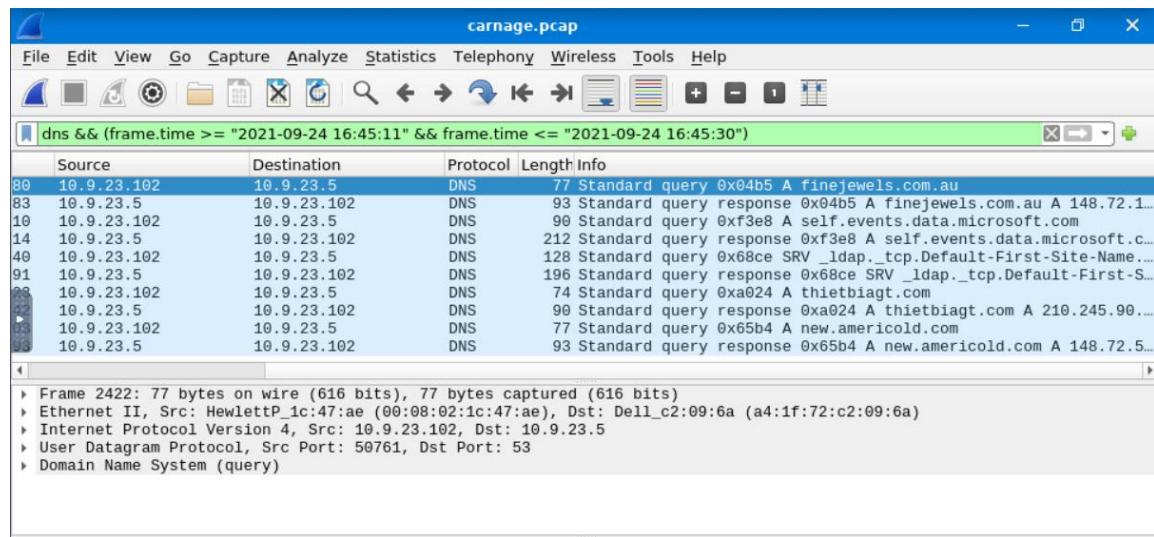
Let's examine the screenshot from the previous task. Answers can be found under the fields "Server" and "x-powered-by" correspondingly.

Answers: LiteSpeed, PHP/7.2.34

Malicious files were downloaded to the victim host from multiple domains. What were the three domains involved with this activity?

Let's examine DNS traffic to solve this task by applying dns as the filtering command. THM provides a query: Narrow down the timeframe from 16:45:11 to 16:45:30. Taking it into account, so the updated filter would be:

dns && (frame.time >= "2021-09-24 16:45:11" && frame.time <= "2021-09-24 16:45:30")



The screenshot shows the Wireshark interface with a packet list window. The filter bar at the top says "dns && (frame.time >= "2021-09-24 16:45:11" && frame.time <= "2021-09-24 16:45:30")". The packet list shows several DNS queries from various sources (10.9.23.5, 10.9.23.102) to destination 10.9.23.5, with responses from finejewels.com.au, self.events.microsoft.com, and thietbiagt.com. The details and bytes panes below the list pane show the structure of the DNS frames.

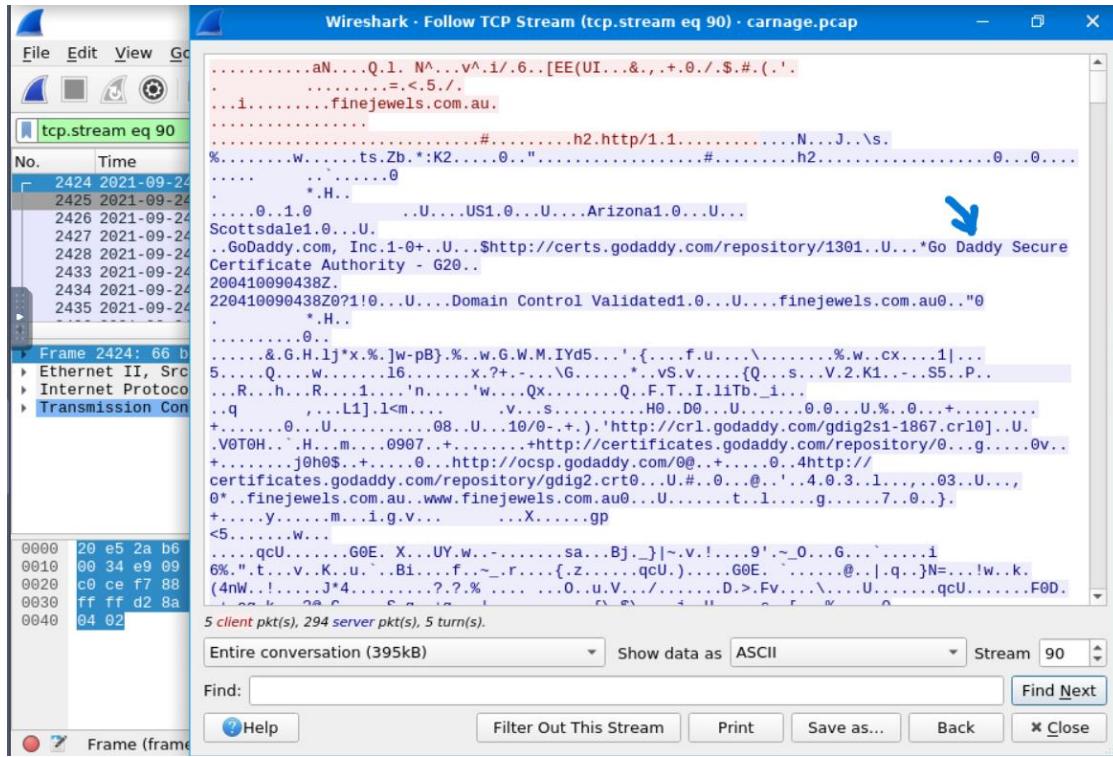
	Source	Destination	Protocol	Length	Info
80	10.9.23.102	10.9.23.5	DNS	77	Standard query 0x04b5 A finejewels.com.au
83	10.9.23.5	10.9.23.102	DNS	93	Standard query response 0x04b5 A finejewels.com.au A 148.72.1...
10	10.9.23.102	10.9.23.5	DNS	90	Standard query 0xf3e8 A self.events.microsoft.com
14	10.9.23.5	10.9.23.102	DNS	212	Standard query response 0xf3e8 A self.events.microsoft.c...
40	10.9.23.102	10.9.23.5	DNS	128	Standard query 0x68ce SRV _ldap._tcp.Default-First-Site-Name...
91	10.9.23.5	10.9.23.102	DNS	196	Standard query response 0x68ce SRV _ldap._tcp.Default-First-S...
93	10.9.23.102	10.9.23.5	DNS	74	Standard query 0xa024 A thietbiagt.com
	10.9.23.5	10.9.23.102	DNS	90	Standard query response 0xa024 A thietbiagt.com A 210.245.90...
	10.9.23.102	10.9.23.5	DNS	77	Standard query 0xb5b4 A new.americold.com
	10.9.23.5	10.9.23.102	DNS	93	Standard query response 0xb5b4 A new.americold.com A 148.72.5...

The output provides a list of the malicious domains participated in the files exchange process with the victim host.

Answer: finejewels.com.au, thietbiagt.com, new.americold.com

Which certificate authority issued the SSL certificate to the first domain from the previous question?

Answer to this question can be obtained by following the TCP stream, done via *Follow > Follow TCP Stream*.



Answer: GoDaddy

What are the two IP addresses of the Cobalt Strike servers? Use VirusTotal (the Community tab) to confirm if IPs are identified as Cobalt Strike C2 servers.

Solving this task requires some statistics collection and analysis. To conduct it, one clicks the tab *Statistics* found in the menu bar, then selects *Conversations*. Let's filter these addresses list by *TCP* and order results by column *Packets*.

After a research, one obtains the needed IP addresses.

Wireshark · Conversations · carnage.pcap									
Ethernet · 8	IPv4 · 109	IPv6	TCP · 447	UDP · 256					
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes
23.111.114.52	65400	10.9.23.102	63557	18002	15M	12281	15M	5721	
10.9.23.102	63555	104.83.84.137	443	9074	10M	1972	107k	7102	
10.9.23.102	63465	185.125.204.174	8080	1375	1387k	408	22k	967	
10.9.23.102	63507	185.106.96.158	80	1074	997k	379	20k	695	
10.9.23.102	63723	177.149.159.181	25	1069	685k	464	652k	605	
10.9.23.102	63439	136.232.34.70	443	1002	989k	284	16k	718	
10.9.23.102	63726	52.97.201.242	25	978	627k	446	592k	532	
10.9.23.102	63610	136.232.34.70	443	976	882k	334	19k	642	
10.9.23.102	63732	52.97.201.210	25	957	616k	441	583k	516	
10.9.23.102	63571	136.232.34.70	443	953	910k	291	16k	662	
10.9.23.102	63736	52.97.201.242	25	945	596k	416	562k	529	
10.9.23.102	63738	217.70.178.9	25	933	585k	405	551k	528	
10.9.23.102	63734	62.149.128.200	25	930	574k	397	540k	533	
10.9.23.102	63747	52.98.163.18	25	922	579k	410	543k	512	
10.9.23.102	63752	185.14.56.240	25	900	573k	392	545k	508	
10.9.23.102	63744	52.97.201.194	25	898	574k	402	541k	496	
10.9.23.102	63694	52.97.201.242	25	886	577k	415	544k	471	
10.9.23.102	63740	45.64.197.254	25	883	573k	300	542k	464	

VirusTotal and the Community tab confirm these are Cobalt Strike C2 servers' addresses:

This IP carried out Apache Log4j RCE attempt(s) (also known as CVE-2021-44228 or Log4Shell). For more information, or to report interesting/incorrect findings, give me a shoutout on @parthmaniar on Twitter.

drb_ra 4 years ago

Cobalt Strike Server Found
C2: HTTPS @ 185[.]106[.]96[.]158:8888
C2 Server: survmeter[.]live/gscpl[.]R/185[.]106[.]96[.]158/gscpl[.]R/
POST URI: /supprq/sa/
Country: United States
ASN: DediPath
Host Header: ocsp[.]verisign[.]com
#c2 #cobaltstrike

Cobalt Strike Server Found
C2: HTTPS @ 185[.]125[.]204[.]174:4444
C2 Server: securitybusinpuff[.]com/jquery-3[.]3[.]1[.]min[.]js,185[.]125[.]204[.]174/jquery-3[.]3[.]1[.]min[.]js
POST URI: /jquery-3[.]3[.]2[.]min[.]js
Country: N/A
ASN: Hydra Communications Ltd
#c2 #cobaltstrike

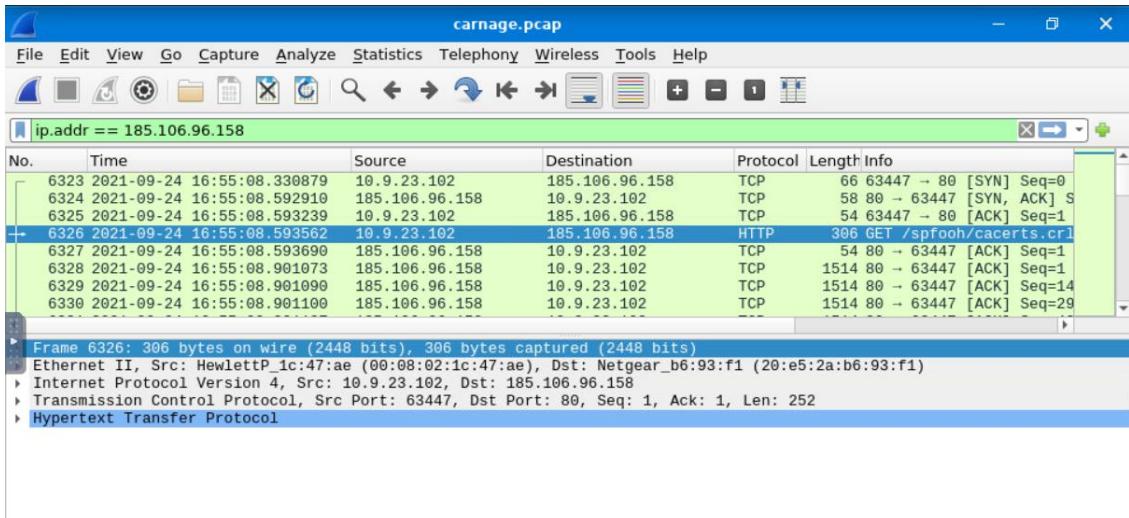
drb_ra 4 years ago

Cobalt Strike Server Found
C2: HTTPS @ 185[.]125[.]204[.]174:8080
C2 Server: securitybusinpuff[.]com/jquery-3[.]3[.]1[.]min[.]js,185[.]125[.]204[.]174/jquery-3[.]3[.]1[.]min[.]js
POST URI: /jquery-3[.]3[.]2[.]min[.]js
Country: N/A
ASN: N/A
#c2 #cobaltstrike

Answer: 185.106.96.158, 185.125.204.174

What is the Host header for the first Cobalt Strike IP address from the previous question?

Now the IP address 185.106.96.158 is known. Let's find the address-related traffic by entering a filtering query `ip.addr == 185.106.96.158`. One is interested in GET methods.



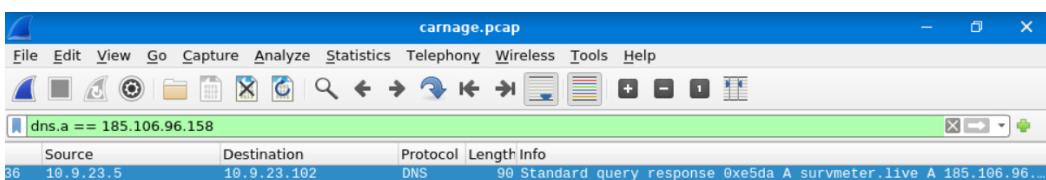
Let's follow its HTTP Stream. The host header is found under the respective field.

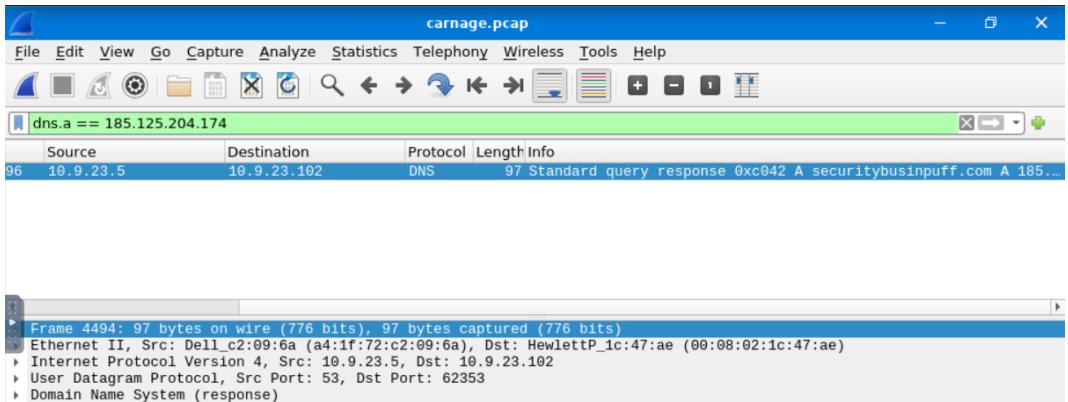


Answer: ocsp.verisign.com

What is the domain name for the first IP address of the Cobalt Strike server? You may use VirusTotal to confirm if it's the Cobalt Strike server (check the Community tab). What is the domain name of the second Cobalt Strike server IP?

There is a hint by THM: Filter out the DNS queries. Let's print out DNS queries for the given addresses by applying the command `dns.a == 185.106.96.158` and `dns.a == 185.125.204.174`:



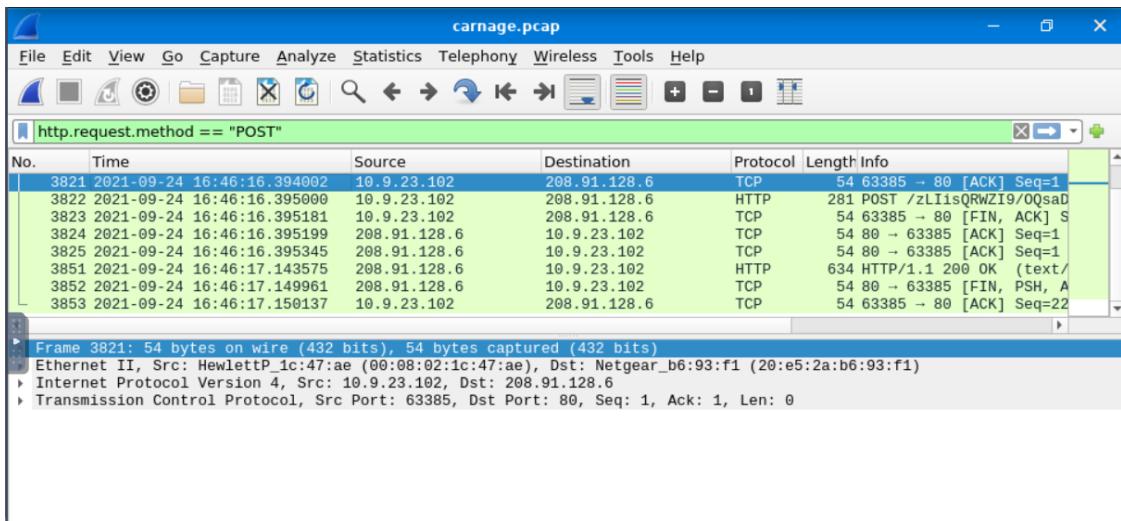


The Info tab discloses the domain names. Taking a look on the re-used screenshots from VirusTotal confirms the findings.

Answers: survmeter.live, securitybusinpuff.com

What is the domain name of the post-infection traffic?

Here THM gives a hint to filter Post HTTP traffic. Applying the filter `http.request.method = "POST"` allows to achieve it.



Following HTTP Stream on the HTTP Post frame 3822 and finding the needed domain name under “Host” field:



Answer: maldivethost.net

What are the first eleven characters that the victim host sends out to the malicious domain involved in the post-infection traffic?

Here one can filter traffic by the malicious domain maldivehost.net from the previous task and reliably obtain the sent characters. Another approach is to simply analyse the sequence of characters which one finds in the followed HTTP Stream for the same frame (the previous screenshot), and subtract the first eleven symbols. It worked for this task.

Answer: zLIisQRWZI9

What was the length for the first packet sent out to the C2 server?

Let's get back to the screenshot with the filtered HTTP Post traffic. Take another look at the first POST packet and observe the value under the column "Length", which is the answer to the question.

No.	Time	Source	Destination	Protocol	Length	Info
3821	2021-09-24 16:46:16.394002	10.9.23.102	208.91.128.6	TCP	54	63385 → 80 [ACK] Seq=1
3822	2021-09-24 16:46:16.395000	10.9.23.102	208.91.128.6	HTTP	281	POST /zLIisQRWZI9/0QsaD
3823	2021-09-24 16:46:16.395181	10.9.23.102	208.91.128.6	TCP	54	63385 → 80 [FIN, ACK] S
3824	2021-09-24 16:46:16.395199	208.91.128.6	10.9.23.102	TCP	54	80 → 63385 [ACK] Seq=1
3825	2021-09-24 16:46:16.395345	208.91.128.6	10.9.23.102	TCP	54	80 → 63385 [ACK] Seq=1
3851	2021-09-24 16:46:17.143575	208.91.128.6	10.9.23.102	HTTP	634	HTTP/1.1 200 OK (text/
3852	2021-09-24 16:46:17.149961	208.91.128.6	10.9.23.102	TCP	54	80 → 63385 [FIN, PSH, A
3853	2021-09-24 16:46:17.150137	10.9.23.102	208.91.128.6	TCP	54	63385 → 80 [ACK] Seq=22

Frame 3821: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
Ethernet II, Src: Hewlett_Packard_1c:47:ae (00:08:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
Internet Protocol Version 4, Src: 10.9.23.102, Dst: 208.91.128.6
Transmission Control Protocol, Src Port: 63385, Dst Port: 80, Seq: 1, Ack: 1, Len: 0

Answer: 281

What was the Server header for the malicious domain from the previous question?

The answer is located under the "Server" field.

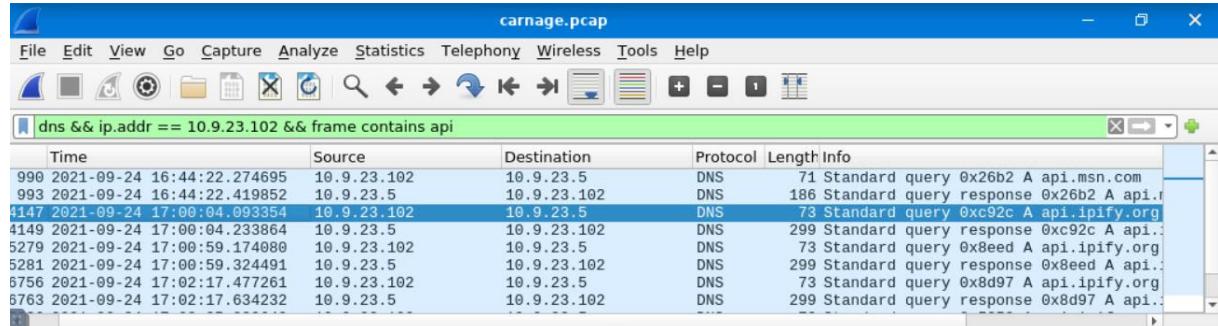
Wireshark - Follow HTTP Stream (tcp.stream eq 104) · carnage.pcap	
POST /zLIisQRWZI9/0QsaDixzHTgtfjMcGypGenp1dWF5eWV9f3k=	HTTP/1.1
Host:	maldivehost.net
Content-Length:	112
Dw8YBxsEGmYFAAEJfR4NQKMMlTYqZDK5KyQm0yRGQqlxEBo4Lzk/EyYrMi1hOT8vIyM7IhcNPzsOKjguFxgkLSIijCxFRgwFAgIIDQUZGBoFD0JFHTTP/1.1	200 OK
Date:	Fri, 24 Sep 2021 16:46:15 GMT
Server:	Apache/2.4.49 (cPanel) OpenSSL/1.1.11 mod_bwlimited/1.4
X-Powered-By:	PHP/5.6.40
Content-Length:	302
Strict-Transport-Security:	...max-age=15552000...
Connection:	close
Content-Type:	text/html; charset=UTF-8

Answer: Apache/2.4.49 (cPanel) OpenSSL/1.1.11 mod_bwlimited/1.4

The malware used an API to check for the IP address of the victim's machine. What was the date and time when the DNS query for the IP check domain occurred? (answer format: yyyy-mm-dd hh:mm:ss UTC)

The IP address of the victim's machine is 10.9.23.102. Let's search for the DNS queries related to the given IP address and find the traffic that includes the usage of API. Applied filter is:

dns && ip.addr == 10.9.23.102 && frame contains api



Time	Source	Destination	Protocol	Length	Info
990 2021-09-24 16:44:22.274695	10.9.23.102	10.9.23.5	DNS	71	Standard query 0x26b2 A api.msn.com
993 2021-09-24 16:44:22.419852	10.9.23.5	10.9.23.102	DNS	186	Standard query response 0x26b2 A api.msn.com
1147 2021-09-24 17:00:04.093354	10.9.23.102	10.9.23.5	DNS	73	Standard query 0xc92c A api.ipify.org
4149 2021-09-24 17:00:04.233864	10.9.23.5	10.9.23.102	DNS	299	Standard query response 0xc92c A api.ipify.org
5279 2021-09-24 17:00:59.174880	10.9.23.102	10.9.23.5	DNS	73	Standard query 0x8eed A api.ipify.org
5281 2021-09-24 17:00:59.324491	10.9.23.5	10.9.23.102	DNS	299	Standard query response 0x8eed A api.ipify.org
6756 2021-09-24 17:02:17.477261	10.9.23.102	10.9.23.5	DNS	73	Standard query 0x8d97 A api.ipify.org
5763 2021-09-24 17:02:17.634232	10.9.23.5	10.9.23.102	DNS	299	Standard query response 0x8d97 A api.ipify.org

Answer: 2021-09-24 17:00:04

What was the domain in the DNS query from the previous question?

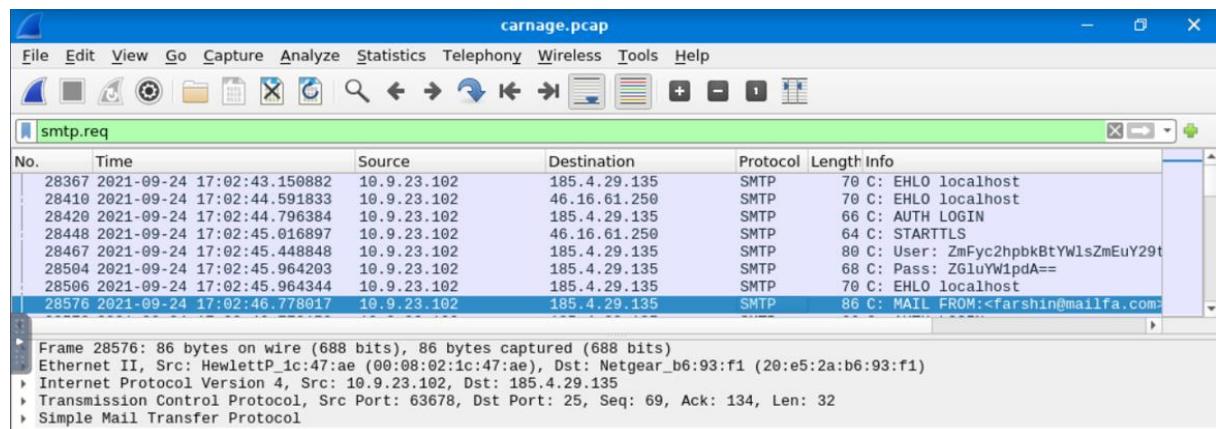
The Info tab discloses the needed domain in the query.

Answer: api.ipify.org

Looks like there was some malicious spam (malspam) activity going on.

What was the first MAIL FROM address observed in the traffic?

The MAIL FROM parameter implies the outgoing traffic, so let's analyse traffic related to the requests and email transfer protocol SMTP by applying filter *smtp.req* and observe the first address containing MAIL FROM:



No.	Time	Source	Destination	Protocol	Length	Info
28367	2021-09-24 17:02:43.150882	10.9.23.102	185.4.29.135	SMTP	70	C: EHLO localhost
28410	2021-09-24 17:02:44.591833	10.9.23.102	46.16.61.250	SMTP	70	C: EHLO localhost
28420	2021-09-24 17:02:44.796384	10.9.23.102	185.4.29.135	SMTP	66	C: AUTH LOGIN
28448	2021-09-24 17:02:45.016897	10.9.23.102	46.16.61.250	SMTP	64	C: STARTTLS
28467	2021-09-24 17:02:45.448848	10.9.23.102	185.4.29.135	SMTP	80	C: User: ZmFyc2hpbkBtYWlsZmEuY29t
28504	2021-09-24 17:02:45.964203	10.9.23.102	185.4.29.135	SMTP	68	C: Pass: ZGluYWlpaA==
28506	2021-09-24 17:02:45.964344	10.9.23.102	185.4.29.135	SMTP	70	C: EHLO localhost
28576	2021-09-24 17:02:46.778017	10.9.23.102	185.4.29.135	SMTP	86	C: MAIL FROM:<farshin@mailfa.com>

Frame 28576: 86 bytes on wire (688 bits), 86 bytes captured (688 bits)
Ethernet II, Src: Hewlett_P_1c:47:ae (00:08:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
Internet Protocol Version 4, Src: 10.9.23.102, Dst: 185.4.29.135
Transmission Control Protocol, Src Port: 63678, Dst Port: 25, Seq: 69, Ack: 134, Len: 32
Simple Mail Transfer Protocol

Answer: farshin@mailfa.com

How many packets were observed for the SMTP traffic?

Let's filter out only SMTP traffic and make use of the built-in functionality of Wireshark, specifically the displayed packets' counter located at the very down:

The screenshot shows the Wireshark interface with a capture file named "carnage.pcap". A filter is applied to show only "smtp" traffic. The packet list pane displays several SMTP sessions between various IP addresses. The details pane shows the structure of a selected SMTP message, including headers like "From" and "To". The bottom status bar indicates there are 70873 packets in total, with 1439 displayed.

Frame 28576: 86 bytes on wire (688 bits), 86 bytes captured (688 bits)
Ethernet II, Src: HewlettP_1c:47:ae (00:08:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
Internet Protocol Version 4, Src: 10.9.23.102, Dst: 185.4.29.135
Transmission Control Protocol, Src Port: 63678, Dst Port: 25, Seq: 69, Ack: 134, Len: 32
Simple Mail Transfer Protocol

0000 20 e5 2a b6 93 f1 00 08 02 1c 47 ae 08 00 45 00 . * G . . E
0010 00 48 2f 4e 40 00 00 06 d3 67 0a 09 17 66 b9 04 H / N @ . . g . . f .
0020 1d 87 f8 be 00 19 c0 8a 3a 88 6c 8b aa ec 50 18 : 1 . . . p .
0030 fa 6b 93 b3 00 00 4d 41 49 4c 20 46 52 4f 4d 3a k . . MA IL FROM:
0040 3c 66 61 72 73 68 69 6e 40 6d 61 69 6c 66 61 2e <farshin@mailfa.com>.
0050 63 6f 6d 3e 0d 0a

Bytes 59-83: Request parameter (smtp.req.parameter) Packets: 70873 · Displayed: 1439 (2.0%) · Marked: 1 (0.0%) · Profile: Default

Answer: 1439

4. Summary

This write-up documents the completion of the Carnage room from TryHackMe. During this challenge I practiced my analytical skills, explored the functionality and usefulness of Wireshark for the network traffic analysis, and put myself into shoes of a network security analyst.

Thank you for your attention!