

- A) 131.247.95.216
- B) Common name: Google, 64.233.161.99, 64.233.161.104, 64.233.161.105
- C) El cliente envía un SYN al puerto 80 para conectarse, es para hacer el handshake, luego el servidor le responde con un ACK el cliente lo recibe y lo reenvía terminando el handshake.
- D) El cliente envía la primera solicitud HTTP el servidor hace un ACK del recibimiento de este paquete por parte del cliente.
- E) Frame 8 es un TCP con la ventana actualizada
- F) El servidor responde con contenido HTTP para el usuario y este le envía un ACK de recepción.
- G) Recibimos otro paquete HTTP por parte del servidor, que permite al cliente continuar con la navegación
- H) Esto sucede debido a que el usuario solicita un servicio mediante el http, el servidor sabe cómo responder a esta solicitud
- I) El servidor le responde al usuario con fragmentos TCP de la solicitud que realizó, el cliente le responde con ACK que confirman la llegada de esta información
- J) El cliente le hace otra solicitud automática, el server responde con un ok y lo envía por el favicon.
- K) El usuario ingresó a google mediante un browser con la intención de hacer solicitud de varias imágenes.

II) Exercise Two

- A) ww.yahoo.akadns.net , 216.109.177.106, 216.109.177.109, 216.109.177.112
- B) En total son 7 paquetes.
- C) Yep, el server utiliza gzip y dentro del header contiene un Set-Cookie
- D) El cliente hizo un request de algo en la página web que se encontraba alojado en otro server, antes de poder jalarlo tiene que resolver el problema con el DNS.
- E) El cliente lo vuelve a solicitar al mismo dominio, sin embargo está solicitando un recurso distinto a un subdominio distinto.
- F) El sistema no necesitaba otro DNS antes debido a que estaba accediendo al mismo, por lo que se volvió a usar este para hacer otros requests.
- G) Los paquetes 141 y 142 no son parte del PDU reensamblado en el paquete 160, sin embargo el paquete 143 sí lo es, cuando se produce que los paquetes llegan en desorden tcp se encarga de solucionar estos problemas.
- H) El cliente hace match a su request usando el TCP Destination port, esto debido a que es un número único, lo que permite que el server le envíe la solicitud correcta al cliente.

III) Exercise Three

- A) Son puertos distintos, el 80 es para el puerto es para la conexión HTTP este puerto no es encriptado, aunque es más veloz, mientras que el puerto 433 es encriptado y sigue el protocolo HTTPS.

- B) La actividad iii es un handshake pero de tipo TLS/SSL, yahoo no tiene esto, podemos ver que my.usf.com uses HTTPS lo cual garantiza la conexión segura, sin embargo yahoo lo hace con HTTP simple, por eso no está listado en esos frames.
- C) Primero el cliente solicita la página root de yahoo con GET / HTTP/1.1, volviendo esto una request general mientras que my.usf.com puede utilizar una ruta más específica que incluya más contexto, debido a la capa extra de seguridad.

IV) Exercise Four

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	8.45.52.227	192.168.0.143	TLSv1.2	78	Application Data
2	0.000000	8.45.52.227	192.168.0.143	TCP	54	443 → 53096 [FIN, ACK] Seq=25 Ack=1 Win=83 Len=0
3	0.000054	192.168.0.143	8.45.52.227	TCP	54	53096 → 443 [ACK] Seq=1 Ack=26 Win=513 Len=0
4	0.077743	8.45.52.227	192.168.0.143	TLSv1.2	78	Application Data
5	0.077743	8.45.52.227	192.168.0.143	TCP	54	443 → 53095 [FIN, ACK] Seq=25 Ack=1 Win=83 Len=0
6	0.077810	192.168.0.143	8.45.52.227	TCP	54	53095 → 443 [ACK] Seq=1 Ack=26 Win=509 Len=0
7	0.949846	192.168.0.113	224.0.0.251	MDNS	77	Standard query 0x0000 PTR _dosvc._tcp.local, "QU" question
8	0.950156	fe80::a231:e87a:c36...	ff02::fb	MDNS	97	Standard query 0x0000 PTR _dosvc._tcp.local, "QU" question
9	2.856422	192.168.0.143	185.125.190.56	NTP	90	NTP Version 4, client
10	3.009732	185.125.190.56	192.168.0.143	NTP	90	NTP Version 4, server
11	3.457062	192.168.0.143	192.168.0.1	DNS	90	Standard query 0x86be A ipv6check-http.steamserver.net
12	3.460892	192.168.0.1	192.168.0.143	DNS	143	Standard query response 0x86be A ipv6check-http.steamserver.net CNAME http.ipv6check.akadns.net CNAME
13	4.774798	162.159.130.234	192.168.0.143	TLSv1.2	105	Application Data
14	4.823292	192.168.0.143	162.159.130.234	TCP	54	52946 → 443 [ACK] Seq=1 Ack=52 Win=512 Len=0
15	4.855164	162.159.130.234	192.168.0.143	TLSv1.2	98	Application Data
16	4.900986	192.168.0.143	162.159.130.234	TCP	54	52946 → 443 [ACK] Seq=1 Ack=96 Win=512 Len=0
17	9.016083	TendaTechnol_d1:0e:...	CloudNetwork_52:5d:...	ARP	60	Who has 192.168.0.143? Tell 192.168.0.1
18	9.016100	CloudNetwork_52:5d:...	TendaTechnol_d1:0e:...	ARP	42	192.168.0.143 is at 50:c2:e8:52:5d:d9
19	9.938523	31.13.67.53	192.168.0.143	SSL	1044	Continuation Data
20	9.986348	192.168.0.143	31.13.67.53	TCP	54	52938 → 443 [ACK] Seq=1 Ack=991 Win=511 Len=0
21	10.005231	192.168.0.143	31.13.67.53	SSL	115	Continuation Data
22	10.058431	31.13.67.53	192.168.0.143	TCP	54	443 → 52938 [ACK] Seq=991 Ack=62 Win=301 Len=0
23	10.847170	192.168.0.143	239.255.255.250	SSDP	167	M-SEARCH * HTTP/1.1
24	11.124340	192.168.0.143	192.168.0.1	TCP	66	53180 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
25	11.124520	192.168.0.143	192.168.0.1	TCP	66	53181 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
26	11.126217	192.168.0.1	192.168.0.143	TCP	54	53 → 53180 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
27	11.126568	192.168.0.1	192.168.0.143	TCP	54	53 → 53181 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
28	11.231405	192.168.0.143	192.168.0.1	TCP	66	53182 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
29	11.231533	192.168.0.143	192.168.0.1	TCP	66	53183 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
30	11.234487	192.168.0.1	192.168.0.143	TCP	54	53 → 53182 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
31	11.234487	192.168.0.1	192.168.0.143	TCP	54	53 → 53183 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
32	11.361088	31.13.67.53	192.168.0.143	SSL	100	Continuation Data
33	11.403412	192.168.0.143	31.13.67.53	TCP	54	52938 → 443 [ACK] Seq=62 Ack=1037 Win=516 Len=0
34	11.442426	192.168.0.143	192.168.0.1	DNS	74	Standard query 0xed9e A www.google.com
35	11.446217	192.168.0.1	192.168.0.143	DNS	90	Standard query response 0xed9e A www.google.com A 142.250.217.196
36	12.418843	192.168.0.189	192.168.0.255	UDP	86	57621 → 57621 Len=44