

1. Respecto al número de mensajes que se envían durante la ejecución del programa, basándose en el esquema de la Figura 1, ¿Cuántos mensajes se deberían enviar durante una partida? Considere ahora su código, ¿Cuántos mensajes se enviaron en una partida?

Según el diagrama debería ser un mínimo de 20 mensajes si es que solo hay 3 jugadas donde se gana en un 3-0, por cada jugada extra se suman 4 mensajes.

Ahora considerando la ejecución de nuestro programa, en Wireshark nos muestra 34 mensajes, esto es debido a los mensajes extras que tenemos por ejemplo el “Handshaking” que se realiza entre el cliente y el servidor al tratarse de una conexión TCP, y además se observa que por cada TCP, existe el doble de mensajes, esto es debido al ACK que se envía, que es una especie de confirmación de que el mensaje fue enviado.

8.635229	127.0.0.1	127.0.0.1	TCP	56 5652 → 55000 [SYN, Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
8.635256	127.0.0.1	127.0.0.1	TCP	56 55000 → 5652 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
8.635268	127.0.0.1	127.0.0.1	TCP	44 5652 → 55000 [ACK] Seq=1 Ack=1 Win=2619648 Len=0
11.849473	127.0.0.1	127.0.0.1	TCP	50 5652 → 55000 [PSH, ACK] Seq=1 Ack=1 Win=2619648 Len=6
11.849490	127.0.0.1	127.0.0.1	TCP	44 55000 → 5652 [ACK] Seq=1 Ack=7 Win=2619648 Len=0
11.851908	127.0.0.1	127.0.0.1	UDP	38 55000 → 50001 Len=6
11.852233	127.0.0.1	127.0.0.1	UDP	34 50001 → 55000 Len=2
11.852336	127.0.0.1	127.0.0.1	TCP	46 55000 → 5652 [PSH, ACK] Seq=1 Ack=7 Win=2619648 Len=2
11.852350	127.0.0.1	127.0.0.1	TCP	44 5652 → 55000 [ACK] Seq=7 Ack=3 Win=2619648 Len=0
23.440447	127.0.0.1	127.0.0.1	TCP	49 5652 → 55000 [PSH, ACK] Seq=7 Ack=3 Win=2619648 Len=5
23.440463	127.0.0.1	127.0.0.1	TCP	44 55000 → 5652 [ACK] Seq=3 Ack=12 Win=2619648 Len=0
23.441183	127.0.0.1	127.0.0.1	UDP	39 55000 → 50001 Len=7
23.441811	127.0.0.1	127.0.0.1	UDP	38 50001 → 55000 Len=6
23.442578	127.0.0.1	127.0.0.1	TCP	150 55000 → 5652 [PSH, ACK] Seq=3 Ack=12 Win=2619648 Len=106
23.442591	127.0.0.1	127.0.0.1	TCP	44 5652 → 55000 [ACK] Seq=12 Ack=109 Win=2619648 Len=0
25.692453	127.0.0.1	127.0.0.1	TCP	50 5652 → 55000 [PSH, ACK] Seq=12 Ack=109 Win=2619648 Len=6
25.692470	127.0.0.1	127.0.0.1	TCP	44 55000 → 5652 [ACK] Seq=109 Ack=18 Win=2619648 Len=0
25.693193	127.0.0.1	127.0.0.1	UDP	39 55000 → 50001 Len=7
25.693771	127.0.0.1	127.0.0.1	UDP	37 50001 → 55000 Len=5
25.694463	127.0.0.1	127.0.0.1	TCP	150 55000 → 5652 [PSH, ACK] Seq=109 Ack=18 Win=2619648 Len=106
25.694478	127.0.0.1	127.0.0.1	TCP	44 5652 → 55000 [ACK] Seq=18 Ack=215 Win=2619392 Len=0
27.059528	127.0.0.1	127.0.0.1	TCP	50 5652 → 55000 [PSH, ACK] Seq=18 Ack=215 Win=2619392 Len=6
27.059551	127.0.0.1	127.0.0.1	TCP	44 55000 → 5652 [ACK] Seq=215 Ack=24 Win=2619648 Len=0
27.060418	127.0.0.1	127.0.0.1	UDP	39 55000 → 50001 Len=7
27.060965	127.0.0.1	127.0.0.1	UDP	38 50001 → 55000 Len=6
27.062297	127.0.0.1	127.0.0.1	TCP	67 55000 → 5652 [PSH, ACK] Seq=215 Ack=24 Win=2619648 Len=23
27.062310	127.0.0.1	127.0.0.1	TCP	44 5652 → 55000 [ACK] Seq=24 Ack=238 Win=2619392 Len=0
27.062463	127.0.0.1	127.0.0.1	UDP	36 55000 → 50001 Len=4
27.062886	127.0.0.1	127.0.0.1	TCP	52 5652 → 55000 [PSH, ACK] Seq=24 Ack=238 Win=2619392 Len=8
27.062895	127.0.0.1	127.0.0.1	TCP	44 55000 → 5652 [ACK] Seq=238 Ack=32 Win=2619648 Len=0
27.062911	127.0.0.1	127.0.0.1	TCP	44 5652 → 55000 [FIN, ACK] Seq=32 Ack=238 Win=2619392 Len=0
27.062917	127.0.0.1	127.0.0.1	TCP	44 55000 → 5652 [ACK] Seq=238 Ack=33 Win=2619648 Len=0
27.062943	127.0.0.1	127.0.0.1	TCP	44 55000 → 5652 [FIN, ACK] Seq=238 Ack=33 Win=2619648 Len=0
27.062957	127.0.0.1	127.0.0.1	TCP	44 5652 → 55000 [ACK] Seq=33 Ack=239 Win=2619392 Len=0

2. Referente a los mensajes realizados por las aplicaciones: ¿Qué tipos de protocolo espera ver? ¿Cuáles encontró? Justifique sus expectativas y las diferencias que encuentre.

Esperaba ver solo TCP/UDP y fue justo lo que se encontró, tal como se ve en la imagen anterior y en la siguiente:

TCP  
TCP  
TCP  
UDP  
UDP  
TCP

3. Las interacciones vía TCP entre el Cliente y el Servidor Intermediario, ¿deben ocupar los mismos puertos a lo largo del tiempo? ¿Coincide con lo visto en Wireshark ? Fundamente.

Protocol	Info
TCP	5652 → 55000
TCP	55000 → 5652
TCP	5652 → 55000
TCP	5652 → 55000
TCP	55000 → 5652

Si, deben ocupar los mismos puertos ya que es una conexión persistente, si coincide esto en wireshark ya que se observa que las interacciones ocurren entre el cliente: 5652 y el servidor 55000.

4. Las interacciones vía UDP entre el Servidor Intermediario y el Servidor Cachipun, ¿Deben ocupar los mismos puertos a lo largo del tiempo?, ¿Que cambia al ejecutar varias partidas seguidas sin cerrar el terminal? ¿Coincide con lo visto en Wireshark ? Fundamente. Si , deben ocupar los mismos puertos a lo largo de una partida, pero si se desea ejecutar varias partidas, existirán distintos puertos entre esas partidas, para esta implementación no se realizó el random de puertos, por lo que se observa que el puerto se mantiene.

UDP	55000 → 50001
UDP	50001 → 55000
TCP	55000 → 5652
TCP	5652 → 55000
TCP	5652 → 55000
TCP	55000 → 5652
UDP	55000 → 50001
UDP	50001 → 55000
TCP	55000 → 5652
TCP	5652 → 55000
TCP	5652 → 55000
TCP	55000 → 5652
UDP	55000 → 50001
UDP	50001 → 55000

5. Respecto al contenido de los mensajes enviados entre cada par de entidades, ¿son legibles? ¿Por qué es o no legible el contenido del mensaje?

Para los mensajes enviados mediante protocolo UDP, los mensajes son legibles, como se ve en la imagen a continuación, esto es debido a que el protocolo no posee cifrado de mensajes:

02 00 00 00 45 00 00 22	75 92 00 00 80 11 00 00	....E.. " u.....
7f 00 00 01 7f 00 00 01	d6 d8 c3 51 00 0e 1a 9c	..... ..Q.....
72 65 61 64 79 3f		ready?

Para los mensajes TCP se puede observar que algunos son ligeramente visibles, esto es debido a que el protocolo encripta de cierta manera los mensajes.

02 00 00 00 45 00 00 2e	75 90 40 00 80 06 00 00	....E.. . u.@.....
7f 00 00 01 7f 00 00 01	16 14 d6 d8 f9 57 d9 82	..... ..W..
b9 c2 34 af 50 18 27 f9	8e 88 00 00 72 65 61 64	..4.P.' . ....read
79 3f		y?