

INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO
Sistemas Distribuidos

Proyecto Alpha

Topos

Integrantes

Jeronimo Aranda Barois - 150344

Diego Villafuerte Soraiz - 142635

14/03/2019

INTRODUCCIÓN

En este proyecto implementado en Java, se ven involucradas muchas técnicas cruciales en el ámbito de los sistemas distribuidos, entre ellas RMI, TCP y UDP.

La invocación remota de métodos de Java(Remote Method Invocation) es un modelo de objetos distribuidos, diseñado específicamente para ese lenguaje, por lo que mantiene la semántica de su modelo de objetos locales, facilitando de esta manera la implantación y el uso de objetos distribuidos. En el modelo de objetos distribuidos de Java, un objeto remoto es aquel cuyos métodos pueden ser invocados por objetos que se encuentran en una máquina virtual (MV) diferente.

Este mecanismo es ocupado para arrancar las sesiones de cada jugador, así como para intercambiar los datos que cada jugador requiere del servidor para establecer las conexiones TCP y UDP posteriores.

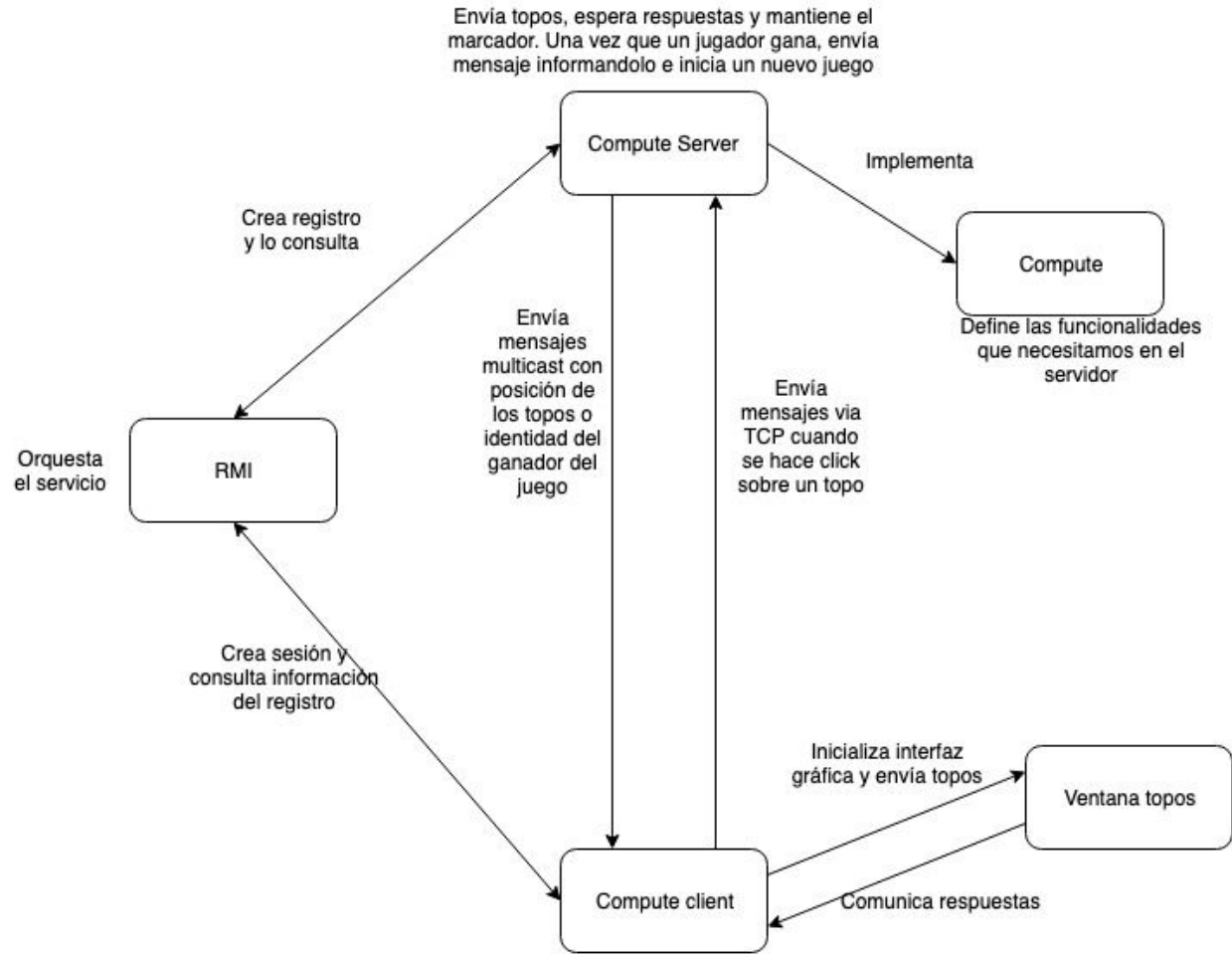
El fin de TCP es proveer un flujo de bytes confiable de extremo a extremo sobre una internet no confiable. TCP puede adaptarse dinámicamente a las propiedades de la internet y manejar fallas de muchas clases. La entidad de transporte de TCP puede estar en un proceso de usuario o en el kernel. Parte un flujo de bytes en trozos y los manda como datagramas de IP. Para obtener servicio de TCP, el emisor y el receptor tienen que crear los puntos terminales de la conexión (los sockets). La dirección de un socket es la dirección de IP del host y un número de 16 bits que es local al host (la puerta). Se identifica una conexión con las direcciones de socket de cada extremo; se puede usar un socket para conexiones múltiples a la vez.

Este mecanismo es usado en el proyecto para establecer conexiones punto a punto entre los jugadores y el servidor para que este sea notificado cuando un jugador ha atacado a uno de los topes.

El protocolo de datagramas de usuario (en inglés: User Datagram Protocol o UDP) es un protocolo del nivel de transporte basado en el intercambio de datagramas (Encapsulado de capa 4 o de Transporte del Modelo OSI). Permite el envío de datagramas a través de la red sin que se haya establecido previamente una conexión, ya que el propio datagrama incorpora suficiente información de direccionamiento en su cabecera. Tampoco tiene confirmación ni control de flujo, por lo que los paquetes pueden adelantarse unos a otros; y tampoco se sabe si ha llegado correctamente, ya que no hay confirmación de entrega o recepción. Su uso principal es para protocolos como DHCP, BOOTP, DNS y demás protocolos en los que el intercambio de paquetes de la conexión/desconexión son mayores, o no son rentables con respecto a la información transmitida, así como para la transmisión de audio y vídeo en tiempo real, donde no es posible realizar retransmisiones por los estrictos requisitos de retardo que se tiene en estos casos.

Este mecanismo es ocupado para enviar los mensajes multicast por parte del servidor hacia cada uno de los jugadores.

ARQUITECTURA



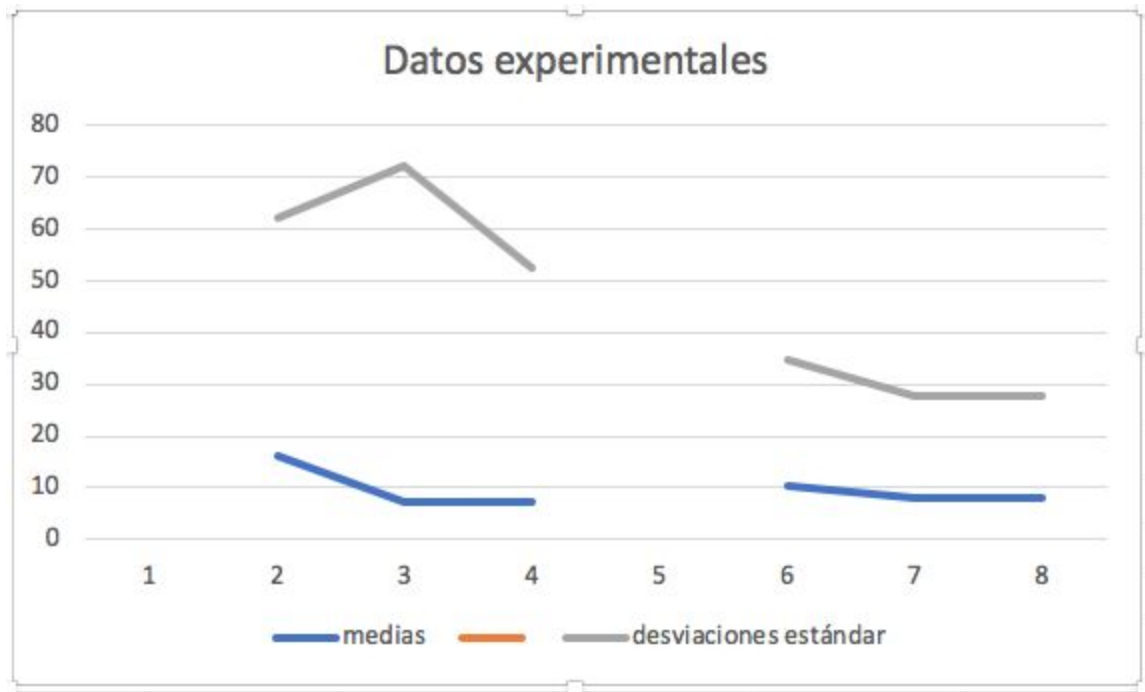
EVALUACIÓN EXPERIMENTAL DEL DESEMPEÑO

En el experimento estresamos el servidor creando 50 y 100 creaciones de sesión del servidor RMI. Esta creación de sesión significa contactar el servicio de RMI y generar una sesión en el lado del servidor, es decir, que el servidor añada la dirección IP del cliente a su lista de jugadores activos, además de recuperar del servidor los puertos y direcciones IP para las posteriores comunicaciones TCP y UDP.

De cada uno de los estresamientos de 50 y 100 creaciones de sesión se hicieron 3 observaciones. Todos los experimentos fueron implementados en una MacBook Air con procesador c1.8 GHz Intel Core i5 y memoria 8 GB 1600 MHz DDR3.

medias	16.06	7.28571429	7.3125	10.39	7.94	7.94
--------	-------	------------	--------	-------	------	------

desviaciones			52.36897	34.77776	27.90236	27.90236
estándar	62.1667967	72.0745079	74	34	22	22



Los resultados fueron convincentes con las hipótesis pues el primer dato siempre es muy grande, al crear la sesión, los posteriores inicios de sesión son mucho más rápidos.

CONCLUSIONES

Trabajar en proyectos de esta complejidad nunca es sencillo. En este caso en particular, nos enfrentamos a varios problemas que rebasan por completo nuestros conocimientos y experiencias. Un buen ejemplo de lo anterior fue cuando intentamos enviar mensajes multicast entre equipos por medio de una red Wi-Fi; lo cual ahora sabemos es imposible dadas las restricciones de seguridad de la red del ITAM. Otro caso representativo fue cuando intentamos ejecutar en nuestras computadoras personales un programa que funcionaba correctamente en las del ITAM y no logramos hacerlo con éxito. Tras horas de investigación, aprendimos que nuestras computadoras estaban configuradas para manejar direcciones IPV6 por defecto y que por ello teníamos que especificar el uso de direcciones IPV4 para poder trabajar con el código existente. A lo largo del desarrollo del proyecto nos enfrentamos a un gran número de situaciones de esta naturaleza, lo que nos ayudó a adquirir nuevos conocimientos, mejorar nuestras habilidades para encontrar respuestas a preguntas desconocidas y en general a consolidar el conocimiento que hemos adquirido en las clases.

Sin embargo, no todo en el desarrollo del proyecto fue frustración, pues al haber sido tan complicado lograrlo, verlo funcionar correctamente fue causa de una gran satisfacción.

REFERENCIAS

- https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/DAAI_RMI.pdf
- <http://neo.lcc.uma.es/evirtual/cdd/tutorial/transporte/tcp.html>