

Tarea 1 de Programación Comunicación de Datos y Redes

Entrega solo a través de Adecca, fecha: lunes 9 de Julio 23:59

Semestre 2017/

Objetivos:

- Alumno aprenda a usar las librerías de java para TCP y UDP dispuestas para establecer transferencia de datos con protocolos de la capa de transporte de la Internet.
- Alumno implemente un protocolo de capa de aplicación sencillo.
- Alumno desarrolle programa cliente/servidor multihebras que implemente el protocolo de comunicación de capa de aplicación solicitado.

Problema:

La empresa Sofanor Video Inc desea que usted le diseñe un protocolo de gestión y de transmisión de capa de aplicación que permita la transmisión de video sobre el protocolo UDP.

Para acceder a este servicio el usuario ejecuta su aplicación cliente, con la cual establece una conexión TCP con la aplicación servidor. Asuma que el servidor puede ofrecer a los menos dos videos llamados "video_1" y "video_2".

El protocolo de solicitud de un video opera del siguiente modo:

1. Cliente envía solicitud por video o mensaje GET <video_#>, donde # es un número mayor o igual a 1. El socket servidor TCP debe ser mayor a 1024.
2. El servidor un mensaje OK si el video existe en caso acompañado por el número de frames del video. Ejemplo: OK 250, lo que significa que el video consta de 250 frames.
3. Si el cliente recibe un mensaje OK, debe abrir un socket UDP en un puerto aleatorio mayor a 1024 y luego enviar al servidor un mensaje PORT #, donde # es el número de puerto seleccionado al azar.

Considere que un video (sin audio) consiste de una secuencia de fotografías o frames. El número total de frames es conocido y definido por usted y es el que debe de haber informado por el servidor al cliente cuando el cliente solicito un video. Seleccione un par de videos desde youtube y descárguelos en formato mp4 o emplee un par de archivos mp4 que usted posea. Existe un sin número de aplicaciones que permiten convertir archivos de video a una secuencia de imágenes. Aquí va un sitio web que describe 5 formas (le recomiendo aquella que emplea VLC)

- <http://www.ogg-converter.net/image/convert-mp4-to-image-picture.htm>
- Enumere las imágenes de una forma conveniente por ejemplo:
 - Video1_1, video1_2, video1_3, etc de modo que le permita automatizar fácilmente la transmisión y recepción de cada frame tanto en el cliente como en el servidor.

Por tanto, **el protocolo de transferencia del video opera del siguiente modo:**

1. Empleando el protocolo UDP, el servidor envía cada una de los frames en un datagrama UDP hacia el cliente, quien está esperando estos datagramas en el puerto que declaro durante el proceso de solicitud de video
2. Cuando el servidor haya transmitido el ultimo frame UDP, debe enviar otro con el mensaje FIN
3. Cuando el cliente recibe un frame debe mostrarlo en un frame java. Mientras el envío sea más rápido mayor será la sensación de estar observando un video.
4. Cuando el cliente reciba el mensaje fin, debe mostrar un mensaje indicando el número de frames recibido y el total.

Comentarios:

- Es obligatorio que su protocolo emplee TCP para la solicitud de video y UDP para la transmisión.
- El servidor debe permitir el envío concurrente de imágenes a distintos clientes, por tanto el **servidor debe operar con hebras**
- Considere videos de dos o 3 minutos a lo más de duración.

Solicitud:

- Debe escribir un informe donde detalle el protocolo diseñado (no más de 2 páginas)
- Incluya códigos fuentes debidamente comentados de sus programas cliente/servidor
- Solo emplee como base los programas java vistos en clases a excepción del código que realice para confeccionar la interfaz gráfica del usuario que permita mostrar las imágenes.
- Captura de pantalla de al menos un paquete que incluya como contenido un tipo de mensaje TCP enviado por su cliente mediante el software Wireshark. Este sniffer de redes puede ser descargado desde:
<https://www.wireshark.org/#download>
- Tutorial básico de Wireshark: https://www.youtube.com/watch?v=vh6q7K_2FHw
- Envío tarea: solo a través de Adecca, adjunte código fuente, informe, archivo README (que explique claramente como compilar y ejecutar tarea) en un solo archivo del tipo tarea_2_2017_1_redes_apellido_alumno1_nombre_alumno1_apellido_alumno2_nombre_alumno2.zip
- El profesor no posee instalado netbeans ni eclipse, de modo que aunque desarrolle su tarea en algún IDE, indique (en archivo README) como compilar y ejecutar su programa por consola/terminal. Se descontaran puntos por el no cumplimiento. El profesor hará el mejor esfuerzo de compilar su tarea pero no la corregirá. Si este paso fallase después de un par de intentos, su tarea será evaluada con la nota mínima.
- LA TAREA PUEDE SER REALIZADA POR DOS ALUMNOS. Ante la detección de copia, queda a discreción del profesor si está será evaluada con nota 1 o se requerirá de una demo explicativa presencial de su tarea por parte del alumno afectado.
- DEBE INSTALAR (COMPILAR Y EJECUTAR) programa servidor en el servidor de alumnos de ICI/IECI al igual que la tarea 1.