Universidad de Valparaíso CHILE

Implementación de servidor TCP

Ignacio Abarca, Felipe Leviñir y Claudio Rojas

Resumen
Se debe implementar un servidor TCP iterativo básico, usando el API Networking en C++. Este debe ser capaz de entregar páginas web básicas y sencillas (Solo texto HTML). En este trabajo se utilizará el avance de la tarea 1, la cual consistió en analizar y capturar el tráfico de datos de una página web mediante Wireshark, los cuales al ser capturados se almacenaron en un archivo txt. Actualmente se requiere de la implementación de un servidor web que sea capaz de mostrar el contenido de una página web.

Palabras clave: TCP / IP, API, servidor web, C ++, HTML, JSON.

Abstract
A basic iterative TCP server must be implemented, using the Networking API in C ++. This should be able to deliver basic and simple web pages (HTML text only). In this work the progress of the task 1 is used, which consists of analyzing and capturing the data traffic of a web page through Wireshark, the files captured in a txt file. Currently requires the implementation of a web server that can display the content of a web page.

Keywords: TCP / IP, API, Web Server, C ++, HTML, JSON.

INTRODUCCIÓN

En este trabajo se implementó un servidor web, a través del lenguaje de programación C++. Este servidor es capaz de servir páginas web sencillas, las cuales deben estar en un texto HTML. El servidor está parametrizado a través de la configuración JSON (interpreta y genera), el cual es un formato ligero de intercambio de datos. Este último es capaz de transmitir la página que el usuario desea ver, en caso que no se logre encontrar la página deseada automáticamente se envía un mensaje de que no se encuentra.

METODOLOGÍA

Inicialmente a partir de la visualización y captura de datos de una página web, se toman aquellos datos, para así compararlos con los datos que se ingresan al servidor y saber si la entrada es correcta. De ser así el servidor responderá al usuario, entregando lo pedido si es encontrado o no.
Para la implementación del servidor, se necesita agregar varias librerías y parámetros, con el fin de planear una forma que logre implementar todas aquellas librerías y parámetros para que el servidor funcione sin ningún problema alguno.

Las herramientas utilizadas dentro del código fueron:

1. JSON(Librería entregada dentro de la tarea): Es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript, Standard ECMA-262 3rd Edition - Diciembre 1999. JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

   JSON está constituído por dos estructuras:
   - Una colección de pares de nombre/valor. En varios lenguajes esto es conocido como un *objeto*, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo.
   - Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.
     Introducción a JSON. (s/f).

2. Json.hpp: Se utiliza para designar los parámetros al servidor.

3. Librería <string.h>:Contiene un conjunto de funciones para manipular cadenas: copiar, cambiar caracteres, comparar cadenas, etc.
   Santos,J. (s/f).

4. Fstream: Transmite la clase tanto para leer como para escribir desde / a archivos.
   Cplusplus. (s/f)

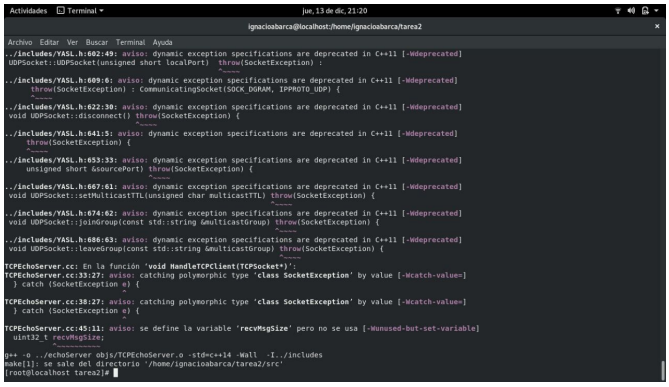
En este trabajo se implementó en el S.O. Ubuntu, la creación del código se realizó en SublimeText.

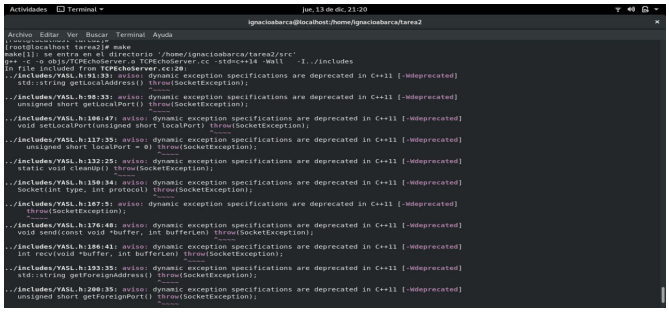
Se utilizaron archivos de textos y todo el trabajo de compilación se hace a través de la consola. Para la realización del código se usó de base lo que ya estaba realizado en la tarea 1, y a partir de ello se empezó a trabajar con el código.

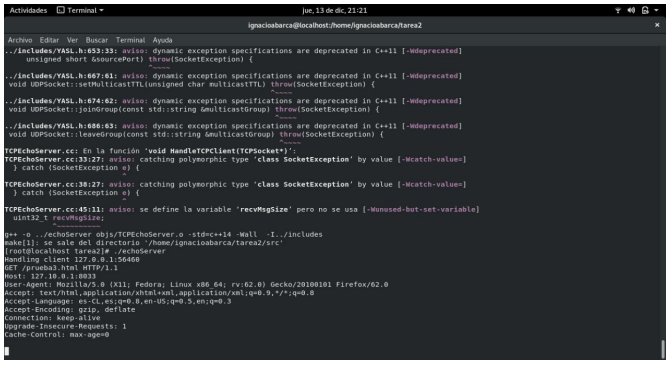
Si la página ingresada es correcta y el servidor la encuentra este mandara este mensaje dependiendo el servidor utilizado, los puertos utilizados son el 8033 y el ip 127.10.0.1.

En caso que la página que el usuario desea ver no la logra encontrar el servidor este le mandara el siguiente mensaje.

El código se compila a través de la consola de la siguiente manera.





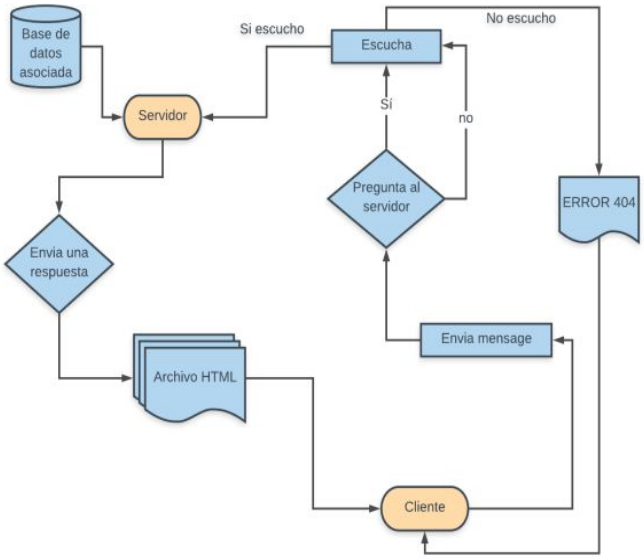


Universidad de Valparaíso CHILE

Vista Lógica:

La estructura del trabajo que se realizó en el código fue la implementación de un servidor TCP el cual fuera capaz de "escuchar" las peticiones del cliente/usuario, si el servidor recibe las datos del cliente, este responde mandando la página web correspondiente, en caso contrario de no recibir nada se enviara un error 404, informando que el servidor no "escucho" nada.
Todo esto se explica en la figura 1.

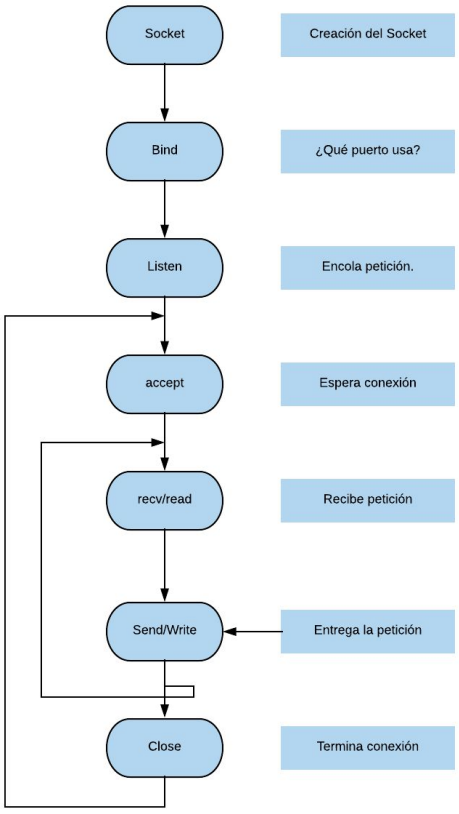
Figura 1.



Vista de Procesos:

Inicialmente se crea el socket para definir los puerto a utilizar, el socket puede intercambiar cualquier flujo de datos que requiera el usuario a la hora de hacer las consultas al servidor, posteriormente el usuario ingresa al puerto que se desea conectar. por ejemplo el puerto 8033, y la IP a utilizar la cual seria la 127.10.0.1 una vez ingresado el usuario entra un tiempo de encolamiento, el cual es cuando el servidor está "escuchando" lo que pide el usuario, tras recibir la petición del cliente el servir entrega la respuesta y termina la conexión, o en el caso de que no termine la conexión el servidor volverá a pedir datos. Como se muestra en la figura 2.

Figura 2.



Universidad de Valparaíso CHILE

REFERENCIAS BIBLIOGRÁFICAS

1. Introducción a JSON. (s/f). Introducción a JSON. Diciembre 12, 2018 , de json Sitio web: https://www.json.org/json-es.html
2. Santos,J. (s/f). Biblioteca de manejo de cadenas (string.h). Diciembre 12, 2018 , de Universidad de las Palmas de Gran Canaria (ULPGC) Sitio web: http://sopa.dis.ulpgc.es/so/cpp/intro_c/introc53.htm
3. Cplusplus. (s/f). Input/output with files. Diciembre 12, 2018, de Cplusplus Sitio web: http://www.cplusplus.com/doc/tutorial/files/