



INSTITUTO TECNOLÓGICO DE COSTA RICA  
INGENIERÍA EN COMPUTACIÓN  
PRINCIPIOS DE SISTEMAS OPERATIVOS

## **Rust The Future Machine**

Dylan González Quesada - 2017118729  
Sebastián Solórzano Guzmán - 2016138505

Profesor:  
Kevin Moraga

**Sede Inter-Universitaria Alajuela  
Abril 2021**

# Tabla de contenidos

<b>1</b>	<b>Introducción</b>	<b>3</b>
1.1	Pre-thread: . . . . .	3
1.2	Pre-forked: . . . . .	3
1.3	StressCMD: . . . . .	3
<b>2</b>	<b>Ambiente de desarrollo:</b>	<b>4</b>
2.1	Entorno de desarrollo: . . . . .	4
2.2	Forma de debuggin: . . . . .	4
2.3	Flujo de trabajo: . . . . .	4
<b>3</b>	<b>Estructura de datos usadas y funciones:</b>	<b>5</b>
3.1	FTPClient . . . . .	5
3.1.1	Funciones . . . . .	5
3.2	FTPServer . . . . .	5
3.2.1	Funciones . . . . .	5
3.2.2	Estructuras . . . . .	5
<b>4</b>	<b>Instrucciones para ejecutar el programa:</b>	<b>6</b>
<b>5</b>	<b>Actividades realizadas por estudiante:</b>	<b>7</b>
<b>6</b>	<b>Autoevaluación</b>	<b>8</b>
6.0.1	Evaluación . . . . .	8
6.0.2	Autoevaluación . . . . .	8
<b>7</b>	<b>Lecciones aprendidas</b>	<b>9</b>
7.1	Dylan . . . . .	9
7.2	Sebastián . . . . .	9
<b>8</b>	<b>Bibliografía</b>	<b>10</b>

# 1 Introducción

Descripción del problema: El problema consiste en administrar mejor los recursos de un servidor FTP por si se llegara a saturar el servicio, para esto se debe realizar un cliente y un servidor web utilizando el protocolo FTP y utilizar las técnicas pre-thread y pre-forked para la administración de los recursos del servidor.

## 1.1 Pre-thread:

Pre-thread es una técnica que consiste en crear previamente varios hilos de ejecución del método que atiende la solicitudes. Estos hilos se crean utilizando la biblioteca pthreads de Unix.

## 1.2 Pre-forked:

Pre-forked es una técnica que consiste en crear previamente varios procesos del método que atiende las solicitudes. Estos procesos se crean utilizando el system call estándar de Unix.

## 1.3 StressCMD:

Adicionalmente se debe crear una aplicación que tendrá como objetivo lanzar un ataque de denegación de servicio sobre los servidores haciendo uso de los clientes FTP, esto con el fin de saturar los servidores hasta que estos no puedan atender más solicitudes.

## **2 Ambiente de desarrollo:**

### **2.1 Entorno de desarrollo:**

Para el desarrollo de la tarea se utilizó el IDE de JetBrains, CLion. En este IDE se trabajó ambos C y Rust.

### **2.2 Forma de debuggin:**

Para el depuramiento se utilizó el debugger que tiene CLion incorporado.

### **2.3 Flujo de trabajo:**

Para el depuramiento se utilizó el debugger que tiene CLion incorporado. Para el control de versiones, se utilizó un repositorio en git, donde habrán 2 ramas, una que sería la rama “master” con la funcionalidad final y una segunda la rama “dev”. En esta segunda es donde se resolvieron todos los issues que se necesitan para resolver el problema.

## **3 Estructura de datos usadas y funciones:**

### **3.1 FTPClient**

#### **3.1.1 Funciones**

main: Se conecta con el servidor y muestra las opciones que se pueden realizar en el servidor FTP.

### **3.2 FTPServer**

#### **3.2.1 Funciones**

push: Función para meter un nuevo nodo a la cola de nodos.

pop: Función para quitar un nodo a la cola de nodos.

error: Función que imprime el error. clientConnection: Función para conectar el servidor con el cliente. Aquí se acepta la conexión y se toma el primer argumento de la llamada y se verifica el tipo de petición que es.

start\_listening: Función para que el servidor escuche. Se pre crean los hilos y el socket que escucha. Luego se establece la dirección y los puertos. Se crea el enlace del socket de escucha a la dirección IP y se verifica.

waitConnection: Esta función es un loop infinito donde se esperan las conexiones, donde se le asigna el cliente al socket que se recibe.

main: Función principal del servidor. Aquí se toman los parámetros de la entrada y se llama a la inicialización y escucha del servidor.

#### **3.2.2 Estructuras**

node: Estructura básica de un nodo.

node\* top: estructura de un nodo con puntero hacia arriba.

node\*base: estructura de un nodo con puntero hacia la base.

## 4 Instrucciones para ejecutar el programa:

Compilar el servidor con el comando:

```
gcc -FTPServer.c -o server
```

Compilar el cliente con el comando:

```
gcc -FTPClient.c -o cliente
```

Ejecutar el servidor con el comando:

```
./server -n 10 -w ../Files -p 8080
```

Ejecutar el cliente con el comando:

```
./cliente 8080
```

## 5 Actividades realizadas por estudiante:

Día	Estudiante	Horas	Actividad
25/04/2021	Dylan	4	Buscar información sobre servidor FTP y programar un poco el
25/04/2021	Sebastián	4	Buscar información sobre pre thread y pre fork
26/04/2021	Dylan	3	Trabajar en el servidor pre thread
26/04/2021	Sebastián	6	Programar cliente en C y servidor pre thread
27/04/2021	Dylan	5	Reunión de equipo y creación de la documentación externa
27/04/2021	Sebastián	5	Reunión de equipo y creación de la documentación externa

## 6 Autoevaluación

Se cuenta con el servidor FTP que utiliza la técnica del pre thread, este inicializa correctamente, crea los hilos y recibe conexiones de clientes, sin embargo no realiza las peticiones del cliente. El cliente está hecho en C y este se conecta correctamente al servidor, envía las solicitudes y recibe la respuesta del servidor. Hubo problemas a la hora de conectar el cliente con el servidor y de procesar las solicitudes del cliente, por lo que no dio tiempo de hacer el servidor pre forked.

### 6.0.1 Evaluación

Calificación	Tema
8	Implementación del Pre-thread
0	Implementación del Pre-forked
0	Servir archivos grandes correctamente
0	Descargar un archivo por FTP desde un navegador u otro cliente FTP
5	Implementación de Protocolos
0	ftplib en Rust
10	ftplib en C
0	Stress-Client
10	Documentación
10	Kick-off

### 6.0.2 Autoevaluación

#### Sebastián

Calificación	Tema
4	Aprendizaje de Pre-thread.
3	Aprendizaje de Pre-forked
4	Aprendizaje de comunicacion entre procesos.
4	Aprendizaje de sockets.

#### Dylan

Calificación	Tema
3	Aprendizaje de Pre-thread.
3	Aprendizaje de Pre-forked
3	Aprendizaje de comunicacion entre procesos.
3	Aprendizaje de sockets.



## **7 Lecciones aprendidas**

### **7.1 Dylan**

Con esta tarea aprendí un poco sobre el funcionamiento de los servidores FTP orientado a hilos. También recordé un poco sobre el lenguaje C y me quedó de lección a administrar mejor el tiempo.

### **7.2 Sebastián**

Como programar un servidor y cliente con el protocolo FTP, el uso de la técnica de diseño de servidores pre thread y en general como trabajar con sockets.

## 8 Bibliografía

Developer, I. (2020). Before you continue to YouTube. Retrieved 25 April 2021, from [https://www.youtube.com/watch?v=7d7\\_G81uews](https://www.youtube.com/watch?v=7d7_G81uews).

Tomar, N. (2020). `nikhilroxtomar/File-Transfer-using-TCP-Socket-in-C`. Retrieved 26 April 2021, from <https://github.com/nikhilroxtomar/File-Transfer-using-TCP-Socket-in-C/blob/master/server.c>

Developer, I. (2020). Before you continue to YouTube. Retrieved 27 April 2021, from <https://www.youtube.com/watch?v=BIJGSQEipEE>