

Trabajo Práctico 1 — File Transfer

Introducción a los Sistemas Distribuidos (75.43)

Curso 02

Primer cuatrimestre de 2025

Alumno	Padrón	Email
Bartocci, Camila	105781	cbartocci@fi.uba.ar
Patiño, Franco	105126	fpatino@fi.uba.ar
Retamozo, Melina	110065	mretamozo@fi.uba.ar
Sagastume, Carlos Matias	110530	csagastume@fi.uba.ar
Sendra, Alejo	107716	asendra@fi.uba.ar

Índice

1. Introducción	2
2. Hipótesis y suposiciones realizadas	3
2.1. Marco teórico	3
2.2. Hipótesis	3
2.3. Suposiciones	3
3. Implementación	4
3.1. Arquitectura del sistema	4
3.2. Reliable Data Transfer	4
3.2.1. Estructuras compartidas	4
3.2.2. Stop & Wait	4
3.2.3. Selective Repeat	4
3.3. Concurrencia	4
3.4. Manejo de errores	4
3.5. Interfaces de usuario	4
3.6. Configuración de parámetros	4
4. Pruebas	5
4.1. Métricas de rendimiento	5
4.2. Herramientas utilizadas	5
4.3. Escenarios de prueba	5
4.4. Resultados obtenidos	5
5. Preguntas a responder	6
6. Dificultades encontradas	7
7. Conclusiones	8

1. Introducción

Contexto del Problema

Objetivos del Trabajo Práctico

Alcance y Limitaciones

Estructura del Informe

2. Hipótesis y suposiciones realizadas

2.1. Marco teórico

2.2. Hipótesis

2.3. Suposiciones

3. Implementación

3.1. Arquitectura del sistema

3.2. Reliable Data Transfer

3.2.1. Estructuras compartidas

Creación de paquetes

Numeración de secuencia

Acknowledgments

Timers y retransmisiones

3.2.2. Stop & Wait

3.2.3. Selective Repeat

3.3. Concurrencia

3.4. Manejo de errores

3.5. Interfaces de usuario

3.6. Configuración de parámetros

4. Pruebas

4.1. Métricas de rendimiento

Para evaluar el rendimiento de la implementación y poder comparar ambos protocolos, se han utilizado las siguientes métricas:

- **RTT promedio:** Tiempo entre el envío de un paquete y la recepción de su acuse de recibo.
- **Jitter:** Desviación estándar del RTT.
- **Throughput:** Cantidad de bytes recibidos sobre el tiempo transcurrido.
- **Overhead:** Cantidad de retransmisiones sobre la cantidad de paquetes enviados.
- **Uso promedio de las ventanas de envío y recepción:** Promedio de tamaño de las ventanas de envío y recepción utilizadas durante la transmisión.

4.2. Herramientas utilizadas

Se utilizó la herramienta *mininet* para simular una LAN simple con tres hosts conectados a través de un switch central. El primer host simula un cliente con una red problemática (10% de pérdida de paquetes). Los dos primeros hosts están limitados a un ancho de banda de 10 Mbps y a una cola de hasta 1000 paquetes, mientras que el tercer host simula un servidor con una red estable y sin limitaciones de ancho de banda.

Para el cálculo de las métricas mencionadas en la sección anterior, se empleó una estructura de datos que permite almacenar la información necesaria para cada una de ellas. Esta estructura se encuentra en el archivo `lib/stats/stats_structs.py` y fue utilizada tanto por el servidor como por los clientes.

4.3. Escenarios de prueba

Para evaluar el rendimiento de los protocolos implementados, se generó un conjunto de archivos de prueba con tamaños entre 1 KB y 10 MB. Cada archivo fue transferido utilizando ambos protocolos (Stop & Wait y Selective Repeat) en las condiciones de red simuladas por *mininet*.

Cada prueba se repitió cinco veces por archivo y protocolo, y se registraron las métricas descritas previamente. Además, las pruebas se llevaron a cabo tanto de forma secuencial como concurrente. En este último caso, se enviaron múltiples archivos de manera simultánea desde diferentes clientes.

4.4. Resultados obtenidos

5. Preguntas a responder

- Describa la arquitectura Cliente-Servidor
- ¿Cuál es la función de un protocolo de capa de aplicación?
- Detalle el protocolo de aplicación desarrollado en este trabajo.
- La capa de transporte del stack TCP/IP ofrece dos protocolos: TCP y UDP. ¿Qué servicios proveen dichos protocolos? ¿Cuáles son sus características? ¿Cuándo es apropiado utilizar cada uno?

6. Dificultades encontradas

7. Conclusiones