

Tecnológico de Costa Rica
Centro Académico de Alajuela
IC-7602 Redes

Proyecto I | Protocolos

Estudiantes:

Erick Blanco
Marco Madrigal
Kevin Zumbado

Profesor:

Ing. Juan Manuel Sánchez Corrales

Alajuela, 03 de octubre de 2022

Índice

Introducción	3
Diseño	4
Protocolo Utopía	4
Protocolo Stop-and-wait	4
Sliding Window	5
Protocolo Go-Back-N	6
Protocolo Selective-Repeat	7
Análisis de resultados	9
Conclusiones	10
Bibliografía	11

Introducción

Dentro de las redes de computación, las conexiones fiables permiten enviar la información en orden y garantizar que se reciba correctamente.

En el presente trabajo, se vio la necesidad de crear un simulador que contiene seis protocolos de capas de enlace diferentes, estos protocolos son capaces de garantizar una conexión punto a punto confiable entre la máquina A y la máquina B.

Un protocolo de red es un conjunto de reglas que rigen la comunicación entre dispositivos conectados a una red. Estas reglas consisten en instrucciones que permiten que los dispositivos se reconozcan y se conecten entre sí, además de aplicar reglas de formato para que los mensajes se propaguen de manera adecuada de principio a fin. Estas reglas de formateo determinan si los datos fueron recibidos correctamente, si fueron rechazados o si hubo algún tipo de problema en la transmisión de la información.

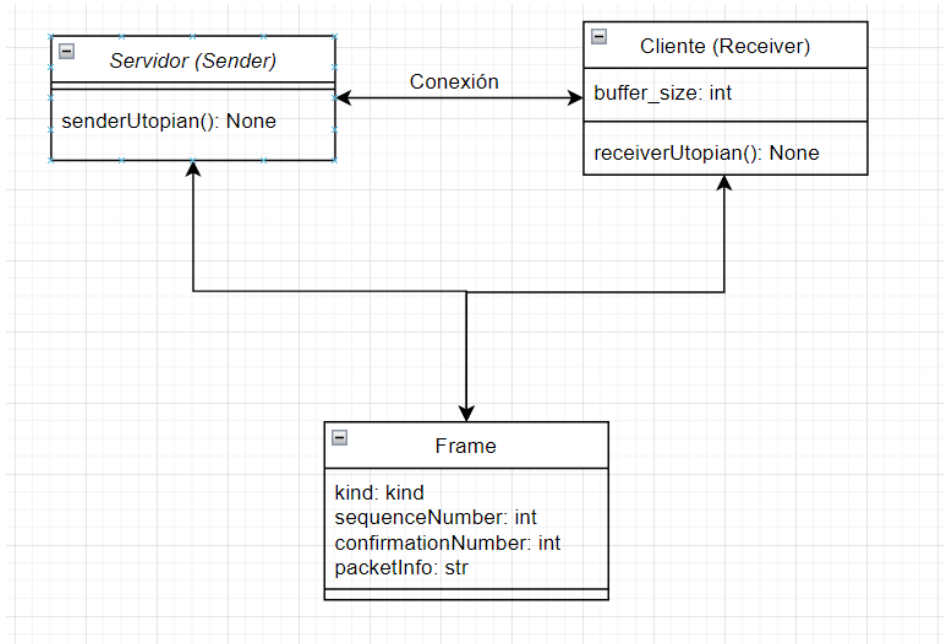
Los protocolos simulados son los siguientes:

1. Protocolo Utopía
2. Protocolo Stop-and-wait:
3. Protocolo PAR
4. Protocolo sliding window de 1 bit
5. Protocolo go-back-n
6. Protocolo selective-repeat:

Diseño

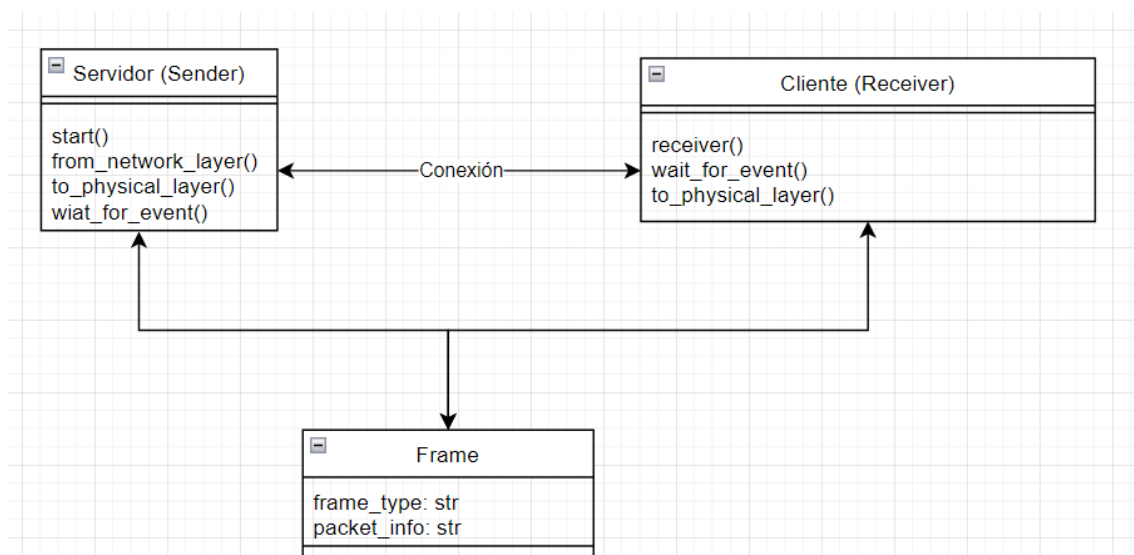
Protocolo Utopía

Este protocolo es considerado como uno de los más simples, se desarrolla bien en ambientes “perfectos”, ya que siempre tiene disposición del canal de red (no comparte) y no existen errores como tal, por ende frame/packet que se envía se debe de suponer que llegará sin errores.



Protocolo Stop-and-wait

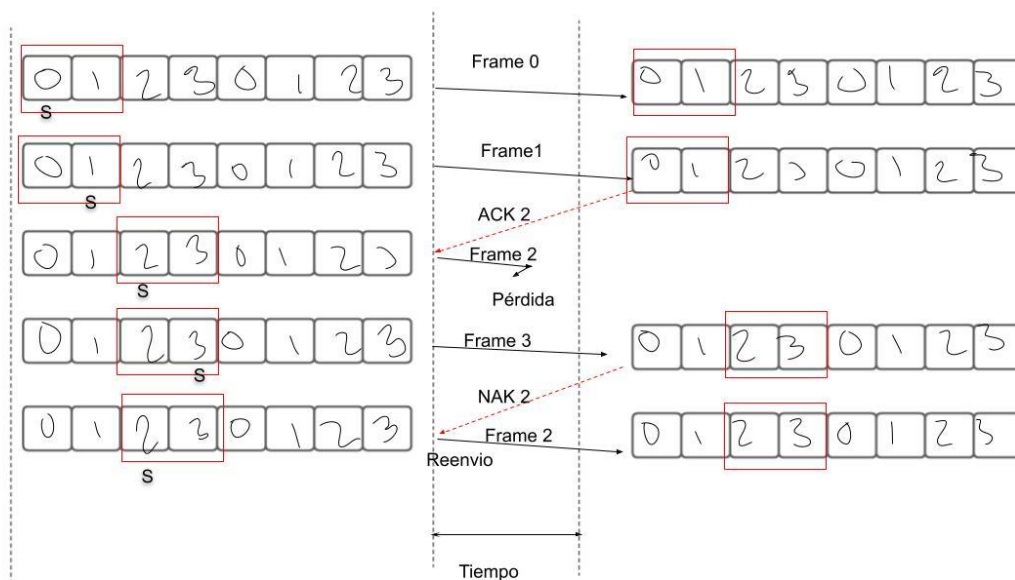
Este protocolo se puede ver como la mejora al protocolo utopía. Su principal intención es incluir una fase de confirmación que se le envía al emisor para que este sepa que puede continuar y volver a emitir más información. Hasta que el emisor no reciba dicha confirmación, no enviará nada ni hará nada.

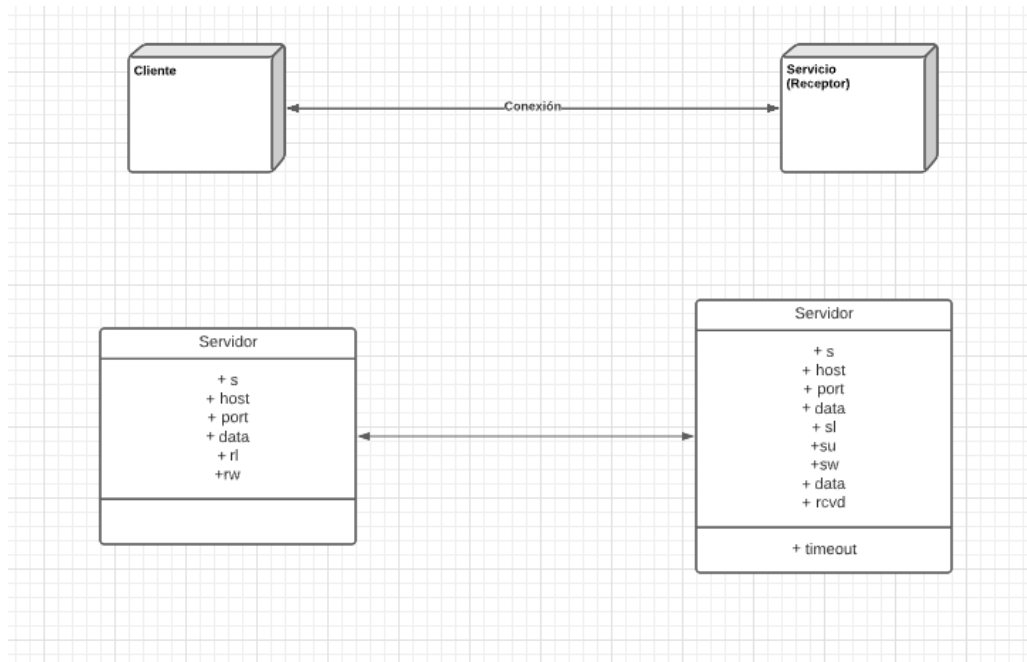


Sliding Window

La ventana deslizante es una técnica para enviar varios fotogramas a la vez. Controla paquetes de datos entre dos dispositivos que requieren una entrega confiable y progresiva de tramas de datos. También se utiliza para TCP (Protocolo de control de transmisión).

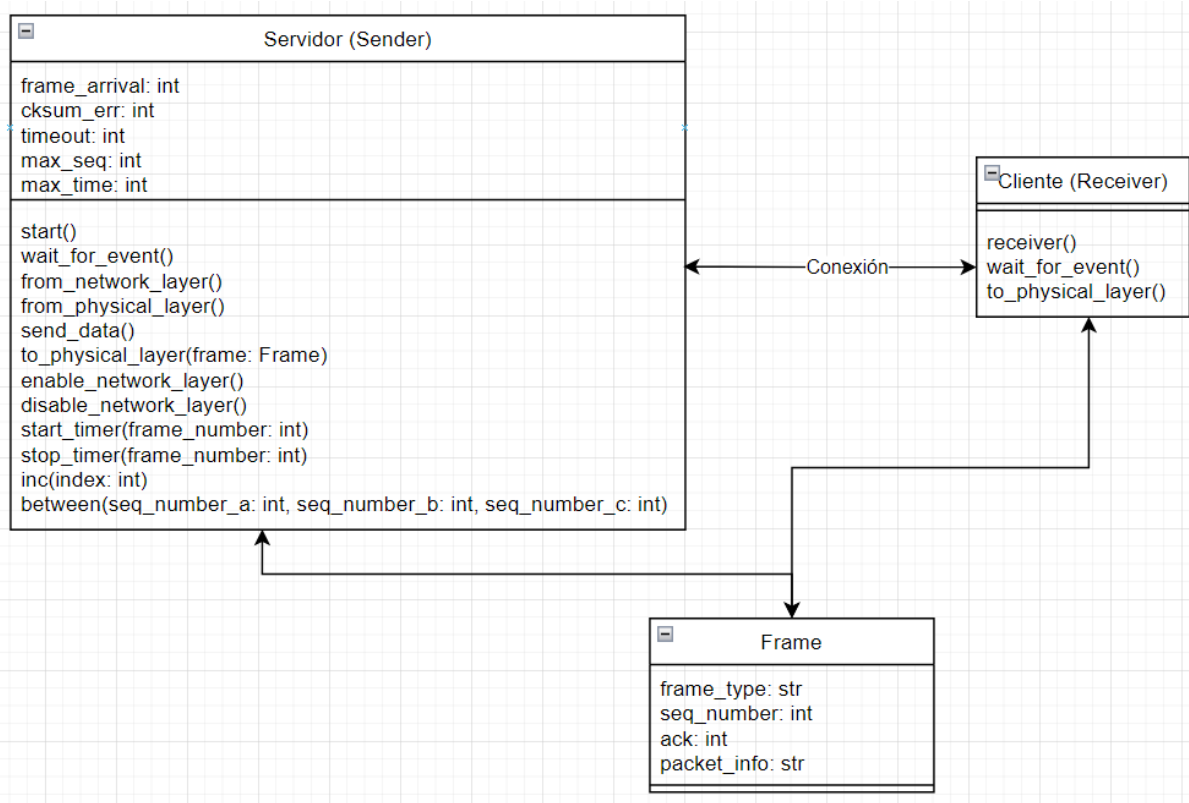
En esta técnica, cada trama se envía desde un número de secuencia. El número de serie se utiliza para encontrar datos faltantes en el extremo receptor. El propósito de la técnica de la ventana deslizante es evitar la duplicación de datos, por lo que utiliza números de secuencia. (Sliding Window Protocol - javatpoint, s. f.)





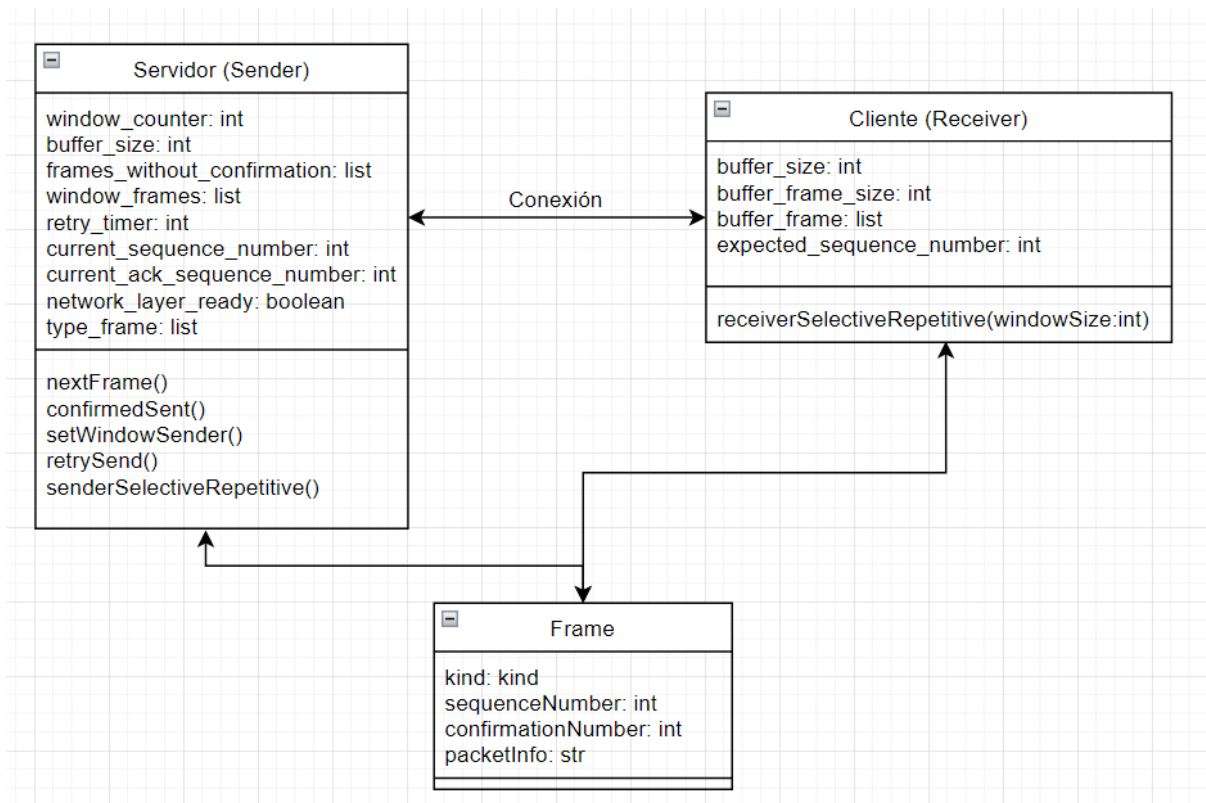
Protocolo Go-Back-N

El protocolo Go-Back-N lo podemos ver como el protocolo de ventana deslizante pero ahora para N cantidad de bits (o "N" como tamaño de ventana emisora). La idea en este protocolo es enviar N frames de una vez al receptor y estar comprobando por las confirmaciones. Estas deben llegar en orden para poder mover la ventana hacia el siguiente frame. Cuando el receptor envía la confirmación del frame y el emisor la recibe correctamente, este procede a mover la ventana y enviar el siguiente frame. En caso de que ocurra un error en este proceso y el emisor no reciba confirmación en cierto tiempo (timeout), procederá a devolverse hasta esa etapa y enviará nuevamente todos los frames a partir de ahí.



Protocolo Selective-Repeat

Es una variante del Go-Back-N (sliding window), consiste en agrupar frames/paquetes en una lista (conocida como ventana) estos son enviados al receptor (la ventana posee las mismas características que el sliding window), en caso de que falle algún envío de estos continúa con el envío de los demás, con la diferencia de que estos no son rechazados sino que son guardados en un buffer en el receptor (esto evita tener que reenviarlos nuevamente comparado con el Go-Back-N) y en cuanto se agote el tiempo de confirmación para el paquete/frame que no pudo ser enviado se reenviará nuevamente y este se acomodará junto con el buffer para mantener el orden de llegada de los paquetes/frames.



Análisis de resultados

En cuanto al aprendizaje de los protocolos, todos los miembros lograron comprender la lógica y el propósito detrás de cada uno y cómo se comportan en una red de verdad, así como la manera en que administran su carga y las diferencias que presentan uno con el otro.

En cuanto a la implementación, la mayoría de los protocolos se lograron desarrollar al 100% logrando así una simulación que permite a cualquier usuario ver cómo se comporta cada protocolo en una red del mundo y qué pasa cuando se exponen a diferentes situaciones de riesgo, como la pérdida de un paquete.

Lamentablemente, por cuestiones de tiempo, alcance y la carga académica por parte de los miembros, la interfaz gráfica no se pudo implementar, por lo que la ejecución de los protocolos presentes se realiza por terminal. En esta misma se muestran mensajes relevantes a la ejecución para lograr comprender el comportamiento de la red.

Conclusiones

En resumen, aprendimos en mayor profundidad el funcionamiento de los protocolos, la complejidad que estos representan (se explican de forma sencilla, sin embargo, a la hora de implementarlo pueden representar un reto), así como las situaciones en los que estos pueden trabajar de forma idónea.

Es interesante ver términos con los que solemos interactuar en nuestra vida diaria, pero que tal vez no conocíamos su funcionamiento o lo que en realidad significan. Ejemplos, como el packet loss en los videojuegos que a menudo suelen salir cuando la conexión está fallando.

Los protocolos son una metodología para garantizar la comunicación, comunicación la cual a veces se puede ver afectada por el entorno, sin embargo, estos aplican medidas necesarias para que la mayoría de información sea enviada y recibida de la mejor manera posible.

Bibliografía

1. *Sliding Window Protocol - javatpoint.* (s. f.). www.javatpoint.com. Recuperado 1 de octubre de 2022, de <https://www.javatpoint.com/sliding-window-protocol>
2. *Protocolos de red: Todo lo que necesitas saber.* (2019, 2 julio). Vester training. Recuperado 1 de octubre de 2022, de <https://vestertraining.com/blog/protocolos-red/>
3. *Go-Back-N Protocol. Baeldung.* Recuperado de <https://www.baeldung.com/cs/networking-go-back-n-protocol>
4. *Stop and Wait ARQ. GeeksforGeeks.* Recuperado de <https://www.geeksforgeeks.org/stop-and-wait-arq/>
5. *Simplex Stop and Wait Protocol. Sharebold.* Recuperado de <https://sharebold.com/posts/simplex-stop-and-wait-protocol-57j9>
6. *A simplex stop and wait protocol for an Error-Free Channel.* Recuperado de http://vikramuniv.ac.in/files/wp-content/uploads/BE_8_SEM_EL_CN-FLOW_CONTROL_PROTOCOL-AMIT_THAKUR.pdf
7. *Go Back N Demonstration.* Recuperado de <https://www.youtube.com/watch?v=gaocB7unrqs>