

Universidad Mariano Gálvez de Guatemala



Facultad de Ingeniería en Sistemas

Curso: Redes de Computadoras II

Proyecto Final: MiniProto – Simulador de Comunicación LAN

Estudiantes: Josué Estuardo Ola Ordoñez

Diego Josué Choj

Esvin Giovanni González de la Cruz

Ing.: NELSON RODOLFO BOCHE SANTOS

Introducción

El proyecto Mini Proto tiene como objetivo principal representar de forma visual y funcional el proceso de transmisión de datos en una red local (LAN), basado en el modelo OSI. Este sistema permite observar cómo los datos son encapsulados, transmitidos y decapsulados a través de las distintas capas, desde la aplicación hasta la física.

Lenguajes y Tecnologías Utilizadas

El proyecto fue desarrollado utilizando tecnologías modernas y de código abierto que permiten simular la comunicación en red y visualizar el flujo de datos de manera didáctica:

- • HTML: estructura las interfaces del Emisor y Receptor.
- • CSS: define el diseño visual de la aplicación.
- • JavaScript: implementa la lógica de la encapsulación y decapsulación en el navegador.
- • Node.js: entorno de ejecución utilizado en el servidor.
- • Express.js: framework que gestiona las rutas HTTP del servidor.
- • WebSocket (ws): permite la comunicación bidireccional en tiempo real entre Emisor y Receptor.
- • JSON: se usa como almacenamiento persistente para guardar el historial de tramas transmitidas.

Persistencia de Datos

El sistema implementa la persistencia mediante un archivo local denominado 'history.json', ubicado dentro del directorio '/server/data/'. Cada transmisión entre el Emisor y el Receptor se registra en este archivo con la información de encabezados, IPs, puertos, tipo de transmisión y contenido enviado. De esta forma, los datos permanecen almacenados, aunque el sistema sea reiniciado.

Funcionamiento General

El flujo general del sistema es el siguiente:

1. El servidor se inicia ejecutando 'npm start' o 'node server.js'.
2. El Emisor ingresa un mensaje, define IP, MAC, puertos y tipo de transmisión desde su interfaz web.
3. Los datos se encapsulan capa por capa siguiendo el modelo OSI y se envían al servidor.
4. El servidor Node.js recibe los datos, los guarda en 'history.json' y los retransmite al Receptor mediante WebSocket.
5. El Receptor decapsula los datos y muestra el mensaje final junto a su versión hexadecimal.

Simulación del Modelo OSI

El simulador Mini Proto representa las capas del modelo OSI de la siguiente manera:

Capa	Función simulada	Ejemplo
Aplicación	Contiene el mensaje del usuario	Hola, Redes!
Transporte	Añade puertos de origen y destino	4000 → 5000
Red	Define direcciones IP	10.0.0.1 → 10.0.0.99
Enlace	Asigna direcciones MAC	02:00:00:00:00:01 → 02:00:00: 00:00:99
Física	Representa el frame en formato hexadecimal	HEX / binario

Resultados y Evidencias

El sistema logró una comunicación exitosa entre Emisor y Receptor mediante WebSocket. El Emisor permite crear y encapsular mensajes, mientras que el Receptor recibe, decapsula y muestra los datos. El servidor central confirma las conexiones activas y mantiene el historial persistente de frames en el archivo JSON.

Conclusiones

El proyecto Mini Proto permite comprender el funcionamiento del modelo OSI de forma práctica y visual. A través del uso de Node.js y WebSocket, se simula una transmisión real de datos entre dispositivos en una red LAN. Además, la persistencia mediante JSON garantiza un registro histórico de las transmisiones. En conjunto, el proyecto refuerza el aprendizaje de las capas de red y su aplicación en entornos reales de comunicación.