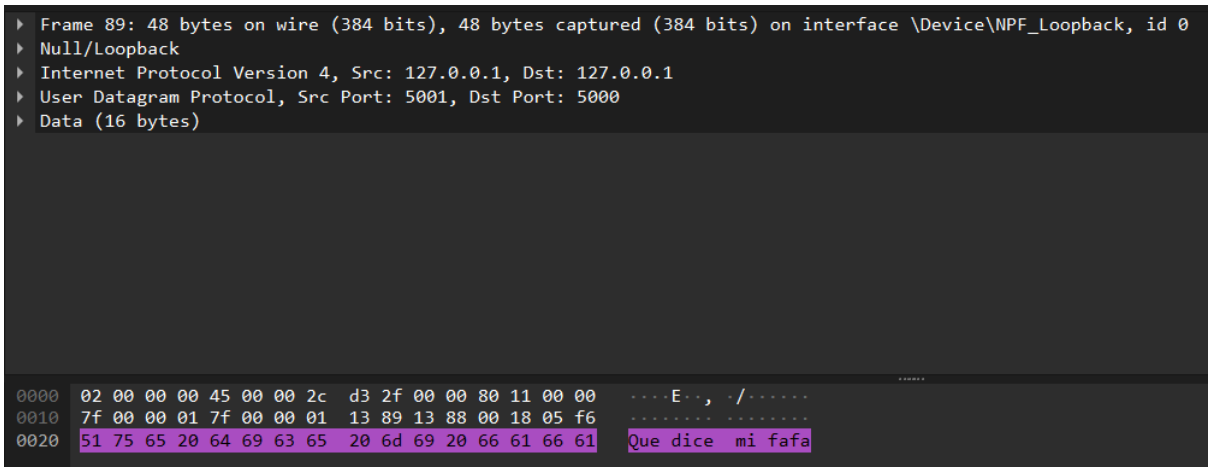
Taller UDP

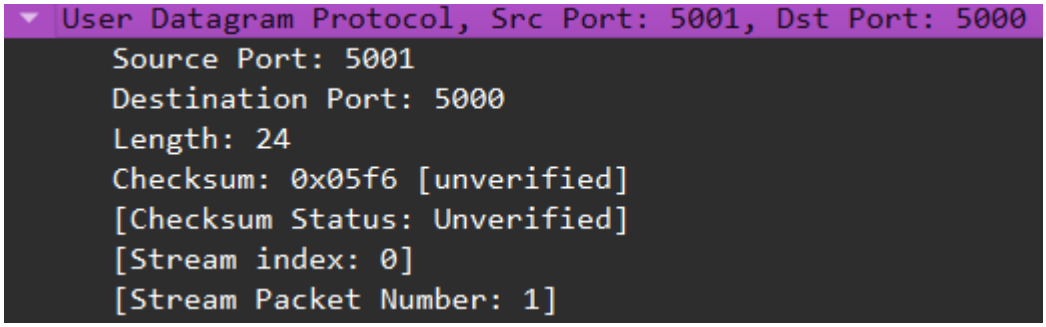
Raul Quigua
Sara Díaz

1) ¿Es posible ver en la captura de Wireshark el contenido del mensaje enviado?



- Sí se puede ver el mensaje, esto se debe a que los mensajes enviados entre los peers no están encriptados, por lo que Wireshark puede capturarlos y mostrar el contenido del mensaje. UDP, a diferencia de algunos otros protocolos, no suele manejar encriptación por sí solo, a menos que usemos una capa adicional (como TLS o DTLS).

2) ¿Cuál es el checksum de la captura? ¿Explique/investiguen porque este checksum?



- 0x05f6 -> 0000 0101 1111 0110 -> 1526
  Este checksum cumple la función de garantizar la integridad del datagrama enviado, en este caso sale como unverified debido a que durante la captura, al estar capturando en la interfaz loopback (local) la cual no verifica checksums, por lo tanto wireshark no puedo verificarlo, sin embargo si revisamos el checksum generado en el emisor del datagrama debería ser el mismo.

3) ¿Qué patrones de diseño/arquitectura aplicaría al desarrollo de un programa basado en red como este?

- Podría ser un patrón Observer, este patrón sería útil para desacoplar la lógica de recepción de mensajes de la interfaz gráfica (en nuestro caso) o cualquier otro componente interesado en recibir notificaciones sobre los mensajes entrantes. En lugar de que UDPConnection tenga una referencia directa a la interfaz gráfica, podríamos implementar el patrón Observer, donde la interfaz se suscribe para recibir constantemente todos los mensajes que se estén enviando.

4) Investiguen que modificaciones son necesarias para implementar este mismo sistema, pero para la comunicación TCP en java

- Diseño del Sistema
    - Servidor
        - Escucha de Conexiones: El servidor debe escuchar conexiones entrantes en un puerto específico.
        - Manejo de Clientes: Una vez que un cliente se conecta, el servidor debe crear un hilo para manejar la comunicación con ese cliente.
        - Broadcast de Mensajes: El servidor debe ser capaz de recibir mensajes de un cliente y reenviarlos a todos los demás clientes conectados.
    - Cliente
        - Conexión al Servidor: Cada cliente debe conectarse al servidor usando su dirección IP y puerto.
        - Envío y Recepción de Mensajes: Cada cliente debe ser capaz de enviar mensajes al servidor y recibir mensajes del servidor.

5) ¿Qué utilidades de codificación o seguridad agregaría al código?

- En este caso como estamos usando UDP, protocolo el cual no posee capa de seguridad, podemos implementar una librería que se llama DTLS (Datagram Transport Layer Security). DTLS es el equivalente de TLS para datagramas y está diseñado para proporcionar seguridad en redes basadas en UDP, puesto que TLS está diseñado para protocolos como TCP. DTLS nos permitiría agregar funcionalidades de encriptación y de manejo de errores en los datagramas en UDP, sin embargo, también tendría un impacto en el desempeño debido a los cifrados y descifrados de datagramas.