

Taller computación internet UDP

Integrantes:

Melissa Hurtado Becerra - A00401116

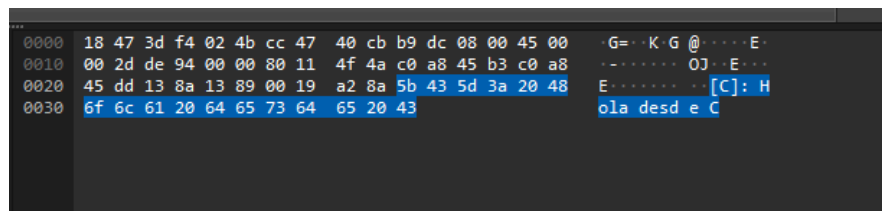
Valentina Tobar Gomez - A00401749

Santiago Grajales Perez - A00402018

Preguntas

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

Si es posible ver en la captura de Wireshark el contenido del mensaje enviado, ya que no estamos usando ningún tipo de encriptación para que no sea posible verlo de manera directa desde el Wireshark.



¿Cuál es el checksum de la captura? ¿Explique/investiguen por qué este checksum?

El checksum de la captura, específicamente en el Frame 17, es 0x47ad.

El valor fue calculado sumando bloques de 16 bits que componen el encabezado UDP, los datos y un pseudo-header que incluye las direcciones IP y otros campos.

En la captura de Wireshark indica el checksum como "unverified" porque no pudo validar la suma debido a que no tiene acceso al pseudo-header completo en este contexto (algo común cuando se analiza tráfico capturado en la misma máquina o cuando la verificación está deshabilitada).

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

Además del patrón que ya usamos para la implementación el cual fue singleton, usaríamos el patrón mediator para el desarrollo de un programa basado en red como este ya que tendríamos la posibilidad de tener dispositivos más independientes. Debido a que los dispositivos ya no necesitan transmitir directamente los mensajes entre ellos sino que pueden hacerlo a través de un componente central encargado de redirigir los mensajes al destinatario correspondiente. Esto da facilidad y ayudaría a que si la red crece y se agregan nuevos dispositivos pueda seguir funcionando sin ningún problema.

Investiguen qué modificaciones son necesarias para implementar este mismo sistema pero para la comunicación TCP en java

Para implementar este mismo sistema de mensajería pero usando comunicación TCP en lugar de UDP, tendría que reemplazar el uso de DatagramSocket por Socket y ServerSocket. Esto requiere de establecer una conexión previa entre los dos peers antes de poder enviar mensajes, y utilizar flujos de entrada y salida (InputStream y OutputStream) en lugar de paquetes.

¿Qué utilidades de codificación o seguridad agregaría al código?

Si se quisiera agregar mayor seguridad al sistema, se podrían implementar mecanismos de cifrado como AES o RSA para proteger los mensajes, así como autenticación entre los peers para verificar la identidad del emisor. Estas mejoras harían el sistema más seguro y robusto, especialmente si se llegara a usar en una red externa o pública.

Ideas: Json