



INF256 - REDES DE COMPUTADORES 2023-02

# ANÁLISIS DE TRÁFICO

## LABORATORIO 1

José Pinto

Ernesto Barría

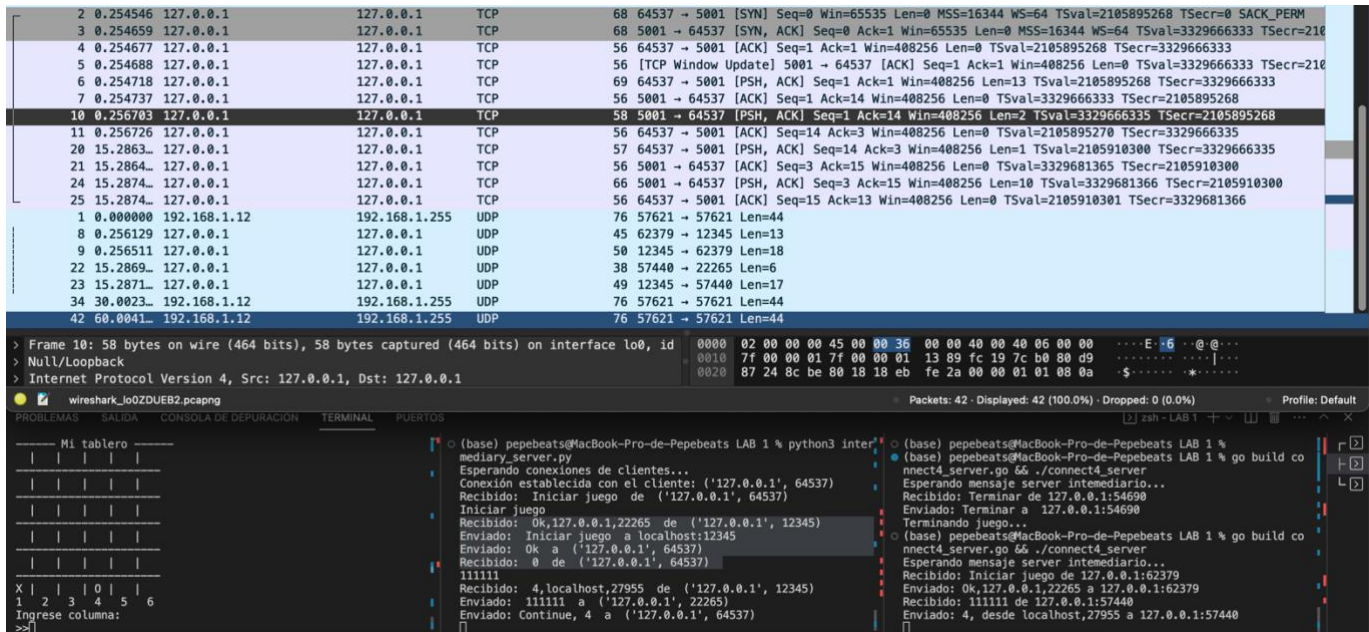
## RESUMEN

En este informe se presenta un análisis exhaustivo del tráfico de red capturado utilizando la herramienta Wireshark. El objetivo de esta evaluación es proporcionar una visión detallada de la comunicación de una aplicación específica a través del análisis de los paquetes de datos intercambiados en la red. A lo largo de este informe, se examinarán las características del tráfico de red, incluyendo el número de mensajes detectados, los protocolos utilizados y la legibilidad del contenido de los mensajes. Este análisis tiene como propósito ofrecer una comprensión completa del comportamiento del juego en términos de comunicación de red.

## DESARROLLO

- 1. Si se analiza el número de los mensajes enviados dentro de la aplicación. ¿Cuántos son los que logra detectar Wireshark? Y comparando en base al código, ¿Es la misma cantidad?, si no lo es, ¿A qué se debería?**

En el análisis realizado, se observa que Wireshark logra detectar la misma cantidad de conexiones UDP que se esperaban en base al código de la aplicación. Esto sugiere que la implementación de UDP en la aplicación es coherente con las expectativas y que Wireshark está capturando de manera efectiva los paquetes UDP generados durante la ejecución. Sin embargo, en el caso del protocolo TCP, se detecta un mayor número de conexiones de las esperadas en comparación con el código de la aplicación. Esta discrepancia podría deberse a varias razones, como la existencia de paquetes TCP adicionales que son formados por el mismo protocolo, por abajo. Otra opción podría ser algunas que otras aplicaciones generen más conexiones TCP en el momento en que se hace el análisis.



Frame 10: 58 bytes on wire (464 bits), 58 bytes captured (464 bits) on interface lo0, id ...  
> Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1

Problemas: Salida: Consola de depuración: Terminal: Puertos

Mi tablero

X | | | | |  
1 2 3 4 5 6  
Ingresar columna:

(base) pepebeats@MacBook-Pro-de-Pepebeats LAB 1 % python3 intermediary\_server.py

Esperando conexiones de clientes...

Conexión establecida con el cliente: ('127.0.0.1', 64537)

Recibido: Iniciar juego de ('127.0.0.1', 64537)

Iniciar juego

Recibido: Ok,127.0.0.1,22265 de ('127.0.0.1', 12345)

Enviado: Iniciar juego a localhost:12345

Enviado: Ok a ('127.0.0.1', 64537)

Recibido: 0 de ('127.0.0.1', 64537)

111111

Recibido: 4,localhost,27955 de ('127.0.0.1', 12345)

Enviado: 111111 a ('127.0.0.1', 22265)

Enviado: Continúe, 4 a ('127.0.0.1', 64537)

(base) pepebeats@MacBook-Pro-de-Pepebeats LAB 1 % go build connect4\_server.go

connect4\_server.go:55: ./connect4\_server

Esperando mensaje server intermediario...

Recibido: Terminar de 127.0.0.1:54690

Enviado: Terminar a 127.0.0.1:54690

Terminando juego...

(base) pepebeats@MacBook-Pro-de-Pepebeats LAB 1 % go build connect4\_server.go

connect4\_server.go:55: ./connect4\_server

Esperando mensaje server intermediario...

Recibido: Iniciar juego de 127.0.0.1:62379

Enviado: Ok,127.0.0.1,22265 a 127.0.0.1:62379

Recibido: 111111 de 127.0.0.1:57440

Enviado: 4, desde localhost,27955 a 127.0.0.1:57440

## 2. ¿Cuál es el protocolo que se debiese ver a la hora de revisar el intercambio de mensajes en Wireshark? ¿Y cuáles encontró?

Al analizar el intercambio de mensajes en Wireshark, se deben revisar los protocolos TCP y UDP, ya que estos son los que componen la aplicación. Los protocolos encontrados fueron TCP, UDP, y mDNS, estos 2 últimos aparecen debido al tráfico de otras aplicaciones o dispositivos en la red.

## 3. ¿El contenido de los mensajes dentro de Wireshark es legible?, ¿Por Qué sí? o ¿Por Qué no?

Wireshark nos entrega información sobre el largo del mensaje, protocolo, fuente y destino, entre otra información, sin embargo, los mensajes no son legibles, están codificados. Esto se debe a los mismos protocolos que se encargan de darle una capa de seguridad a la información codificándolos, ayudando así al manejo de estos.