Prueba – Testeo de Seguridad en Redes de Datos

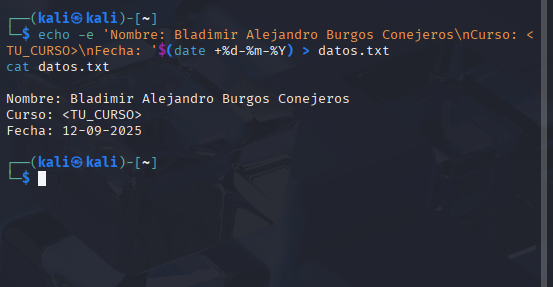
Alumno: Bladimir Alejandro Burgos Conejeros

# Punto 1 – Análisis de tráfico con hping3 (Kali)

Se ejecutaron pruebas con hping3 para ICMP, TCP (puerto 80), UDP (puerto 53) y TCP con payload desde un archivo de texto (datos.txt). Cada prueba se validó con capturas en terminal/Wireshark.

## 1.1 Archivo personal (datos.txt)

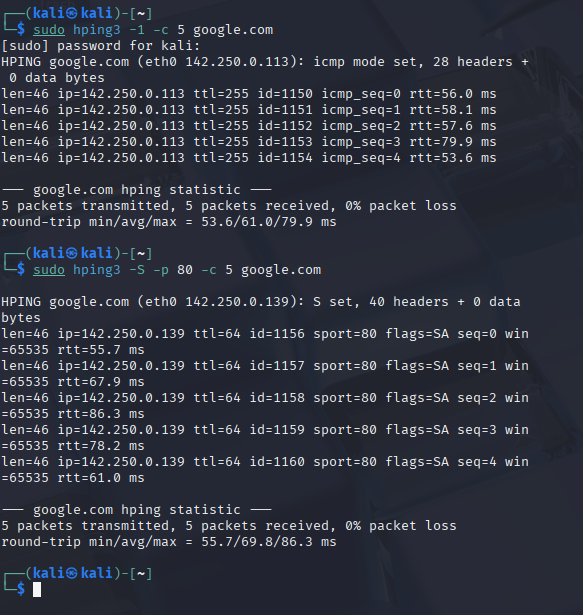
Archivo generado con nombre, curso y fecha (reutilizado como payload en la prueba TCP).



*Evidencia: contenido de datos.txt*

## 1.2 a) ICMP (ping)

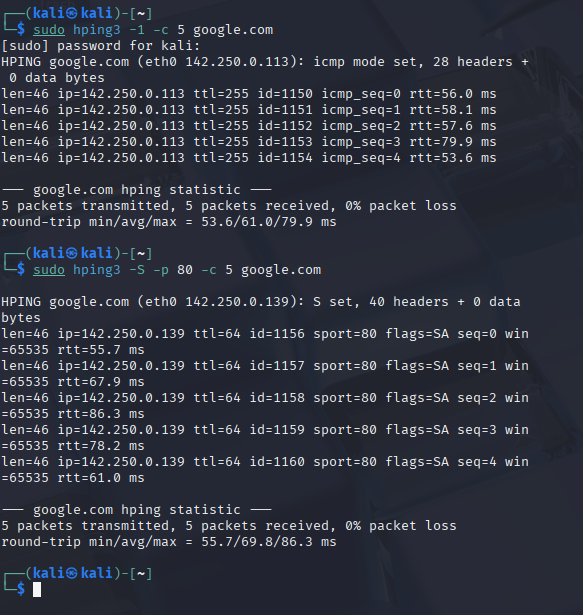
Comando: sudo hping3 -1 -c 5 google.com



*Evidencia: hping3 ICMP (arriba) con 0% pérdida*

## 1.2 b) TCP SYN a puerto 80

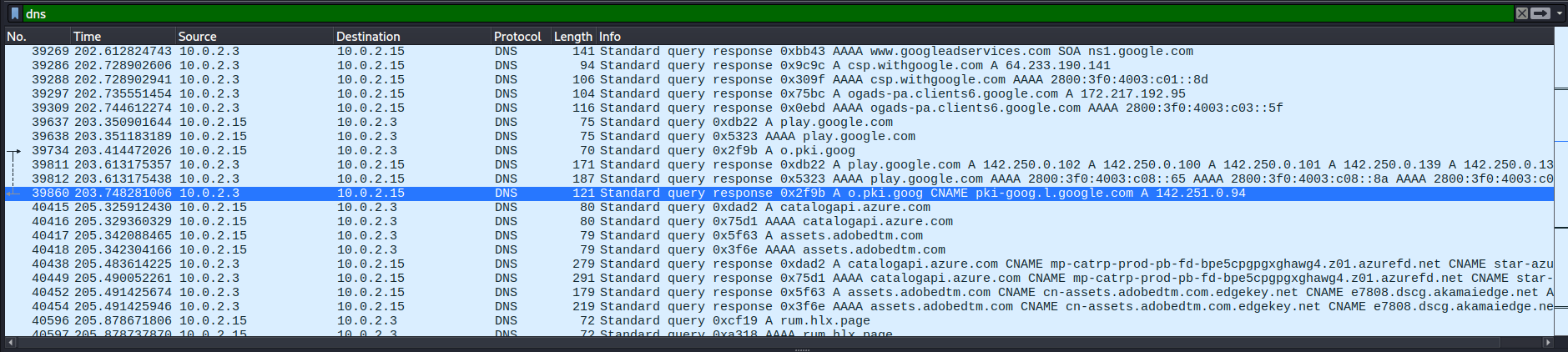
Comando: sudo hping3 -S -p 80 -c 5 google.com



*Evidencia: hping3 TCP SYN a 80 (flags SA = SYN/ACK)*

## 1.2 c) UDP a puerto 53 (DNS)

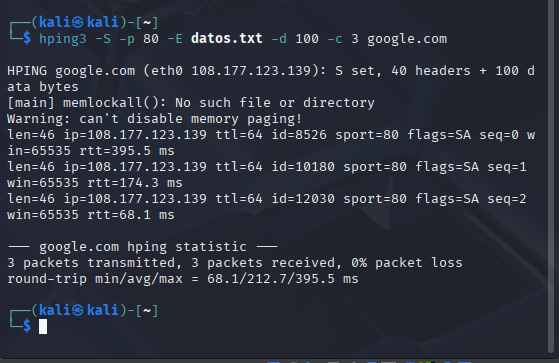
Comando: hping3 --udp -p 53 -c 5 google.com



*Evidencia: DNS (query/response) sobre UDP/53*

## 1.2 d) TCP con payload (datos.txt)

Comando: hping3 -S -p 80 -E datos.txt -d 100 -c 3 google.com

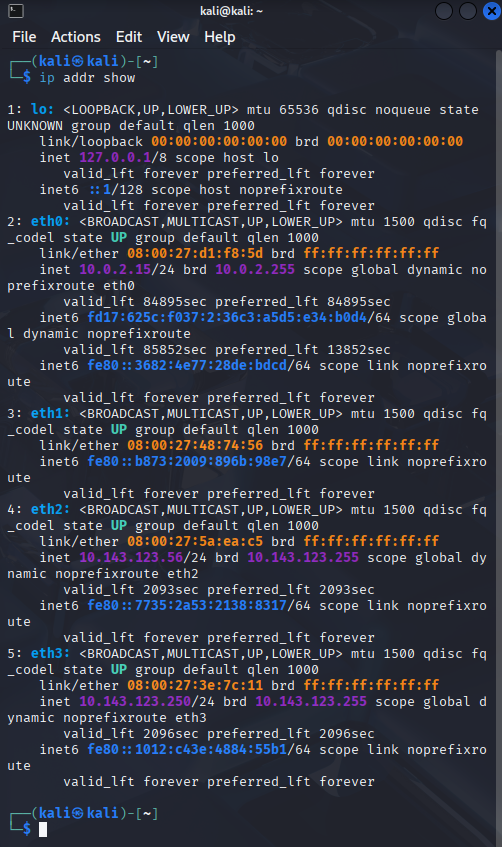


*Evidencia: hping3 enviando payload desde datos.txt*

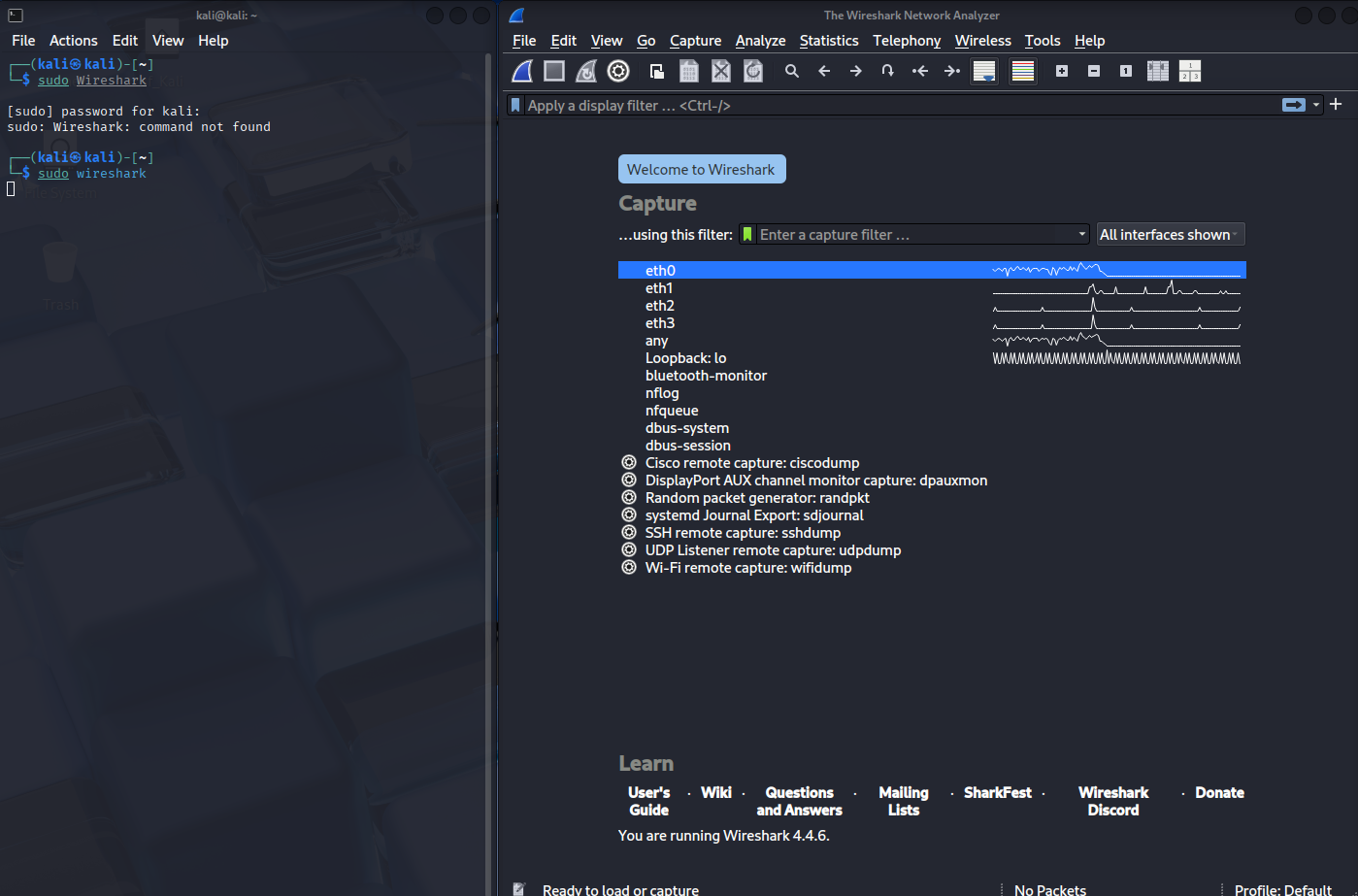
# Punto 2 – Captura y análisis en Wireshark (10–15 min)

Se realizó captura en interfaz eth0 (IP local 10.0.2.15), visitando ≥5 sitios, descargando un archivo y generando tráfico DNS/ICMP. Posteriormente se aplicaron filtros de visualización y se obtuvieron estadísticas.

## 2.1 Interfaz e inicio de captura



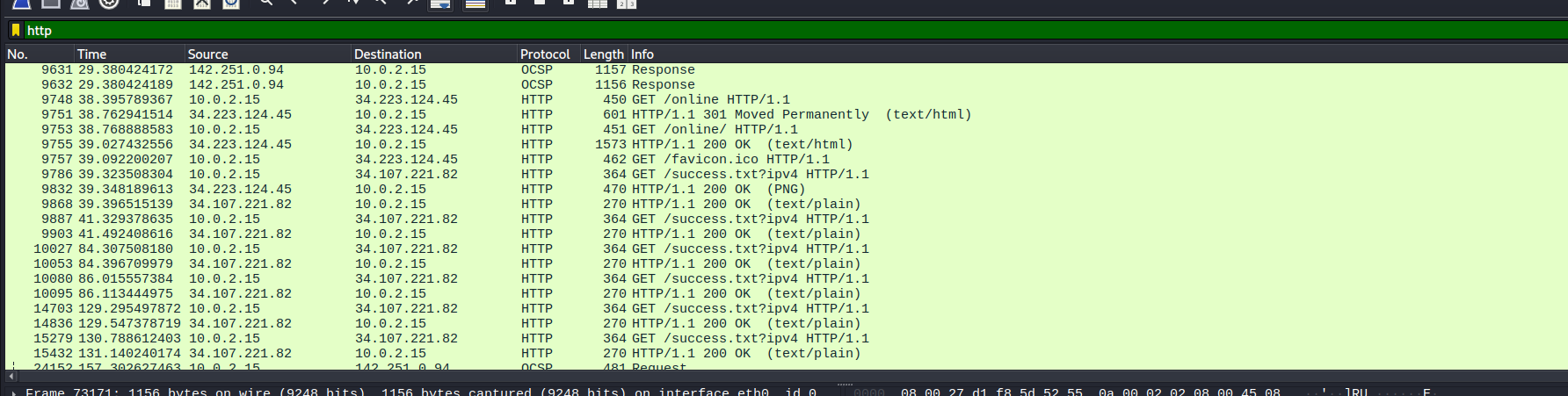
*Evidencia: ip addr show – IP local 10.0.2.15 en eth0*



*Evidencia: Wireshark preparado en interfaz eth0*

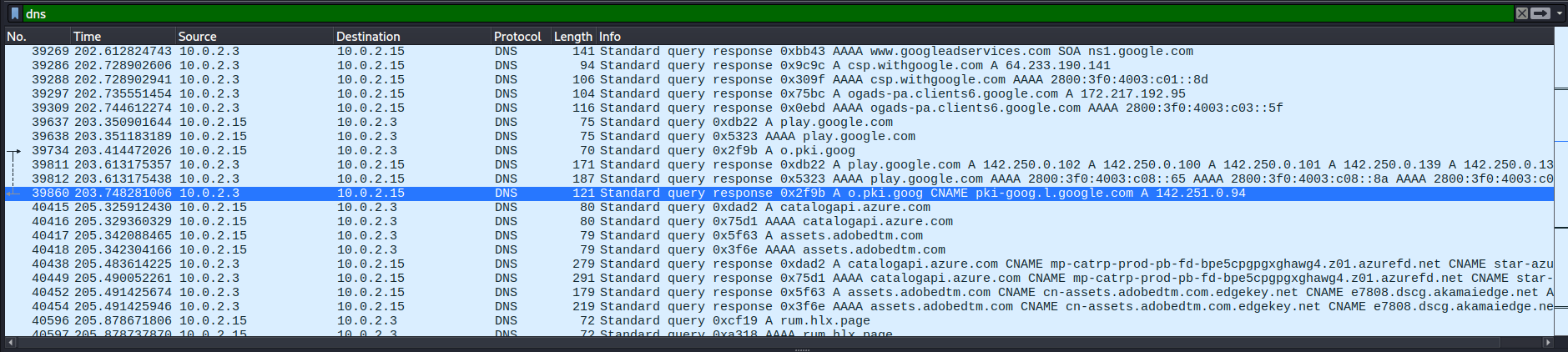
## 2.2 Filtros aplicados y evidencias

a) http – peticiones/respuestas en texto plano



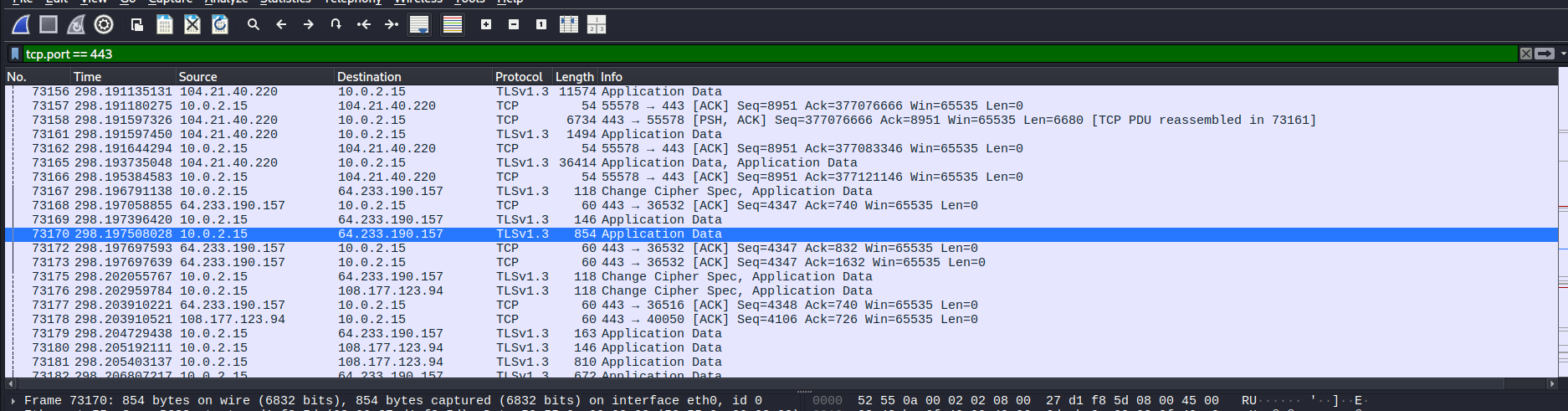
*Evidencia: Filtro http*

b) dns – consultas y respuestas DNS



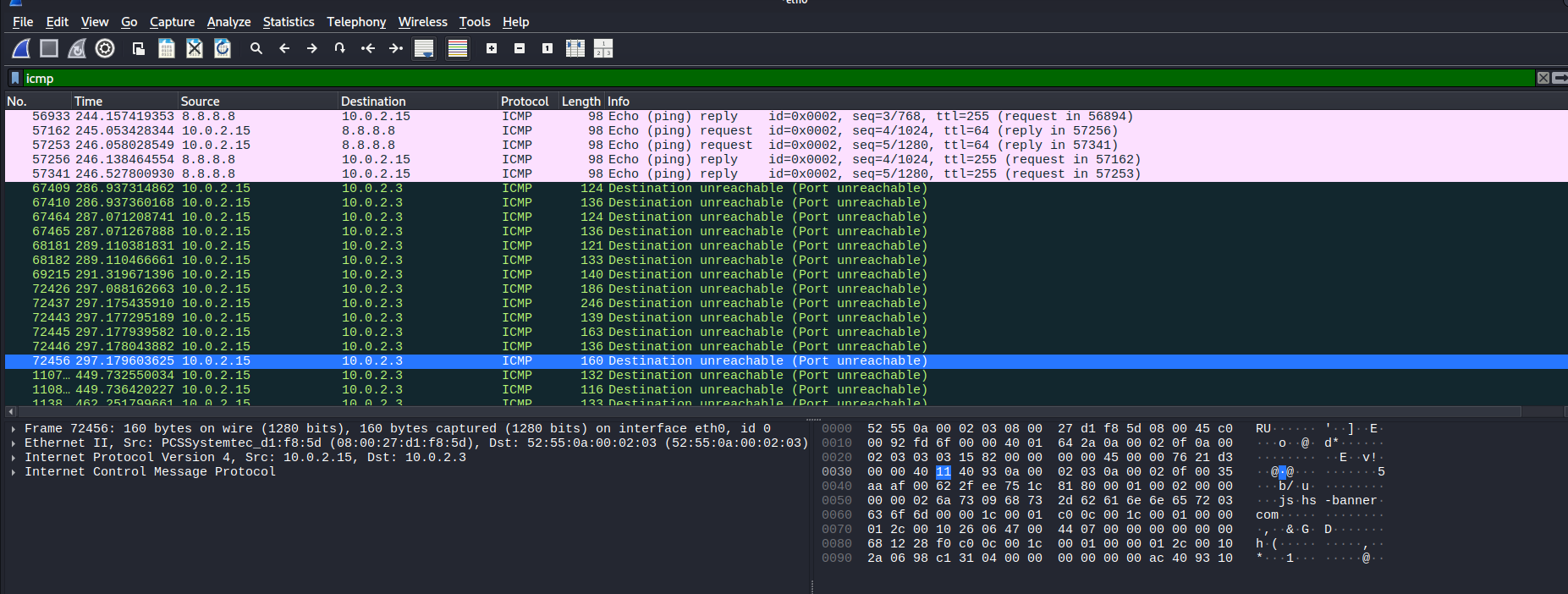
*Evidencia: Filtro dns (registros A/AAAA/CNAME)*

c) tcp.port == 443 – tráfico TLS/HTTPS



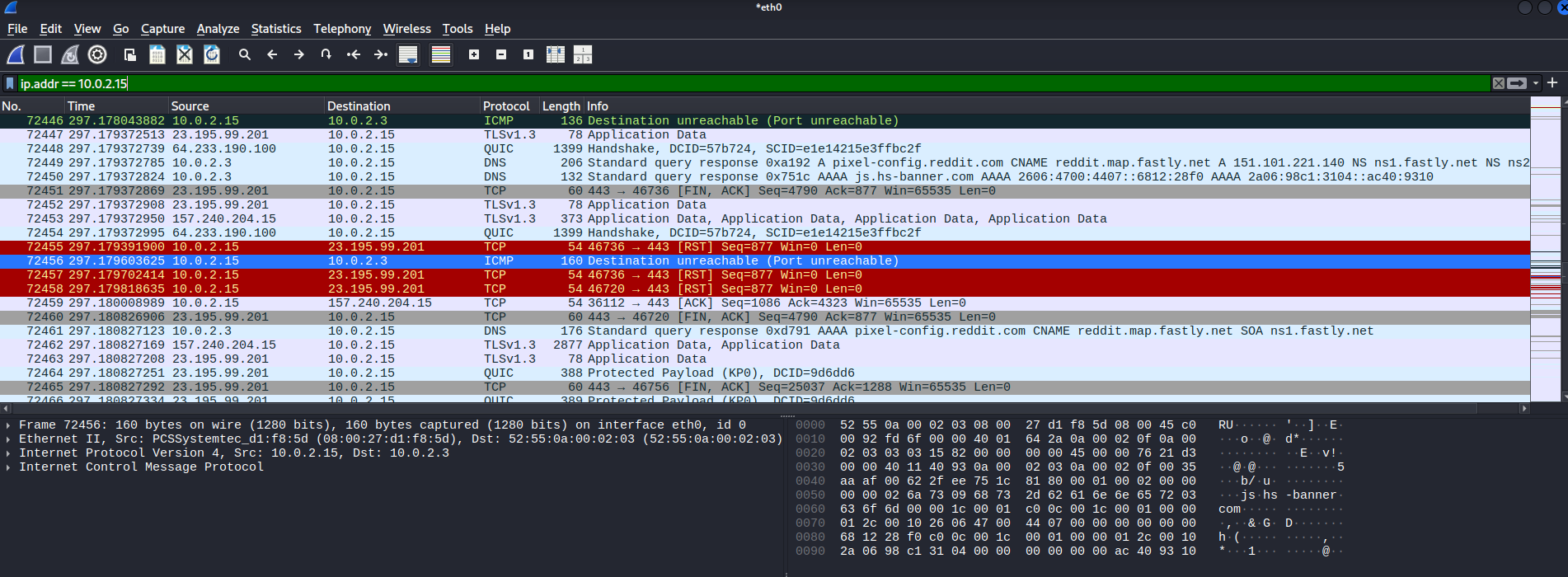
*Evidencia: Filtro tcp.port == 443 (TLS 1.3 Application Data)*

d) icmp – eco request/reply y Destination Unreachable



*Evidencia: Filtro icmp (echo y Port unreachable)*

e) ip.addr == 10.0.2.15 – todo el tráfico del host local



*Evidencia: Filtro ip.addr == 10.0.2.15*

## 2.3 Estadísticas

Protocol Hierarchy – distribución de protocolos en la sesión



*Evidencia: Statistics → Protocol Hierarchy*

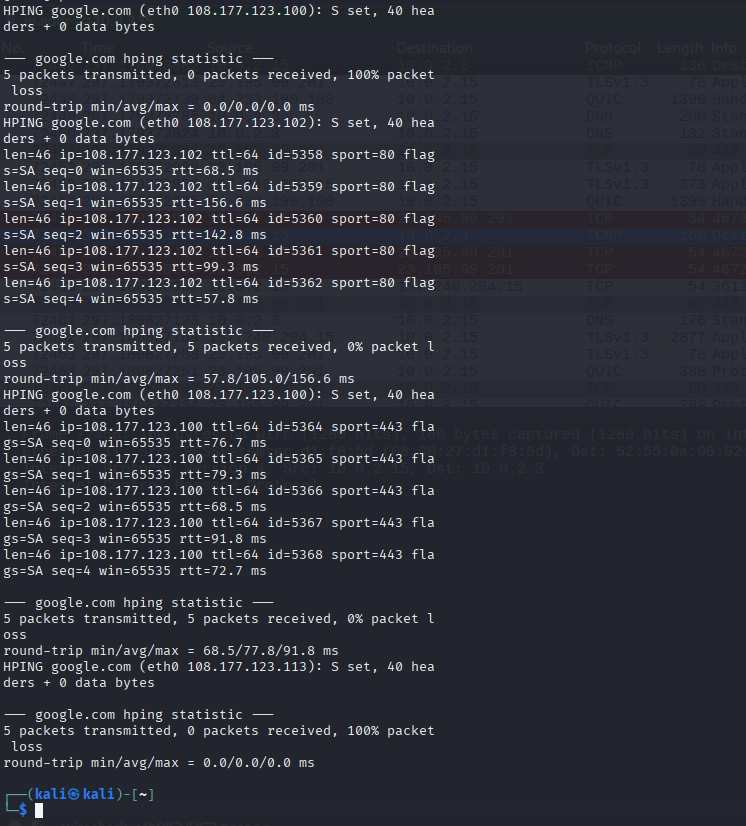
Endpoints (IPv4) – \*pendiente de insertar si se comparte la captura correspondiente\*

# Punto 3 – Conectividad a puertos 22, 80, 443 y 21

Se utilizó hping3 para probar conectividad a los puertos más comunes de servicios de red. Las pruebas se complementaron con capturas en Wireshark para verificar los flags de respuesta.

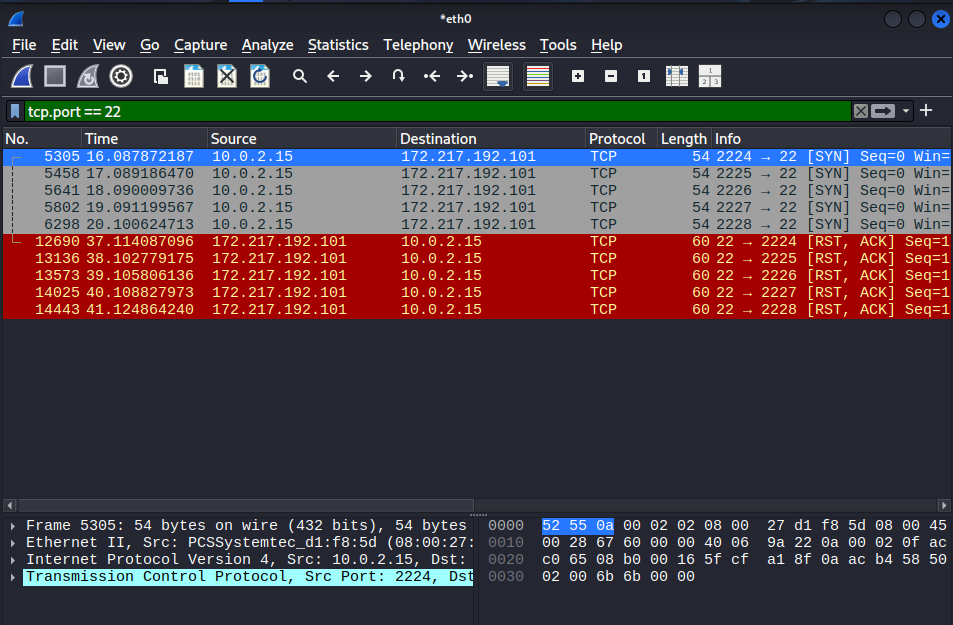
|  |  |  |
| --- | --- | --- |
| **Puerto** | **Estado observado** | **Evidencia** |
| 22 (SSH) | Cerrado (respuesta RST/ACK) | hping3 y Wireshark muestran reset |
| 80 (HTTP) | Abierto (SYN/ACK) | Respuesta positiva, aunque puede redirigir a HTTPS |
| 443 (HTTPS) | Abierto (SYN/ACK) | Establece conexión TLS/SSL cifrada |
| 21 (FTP) | Cerrado (respuesta RST/ACK) | Puerto no disponible en host público |

## 3.1 Resultados en hping3

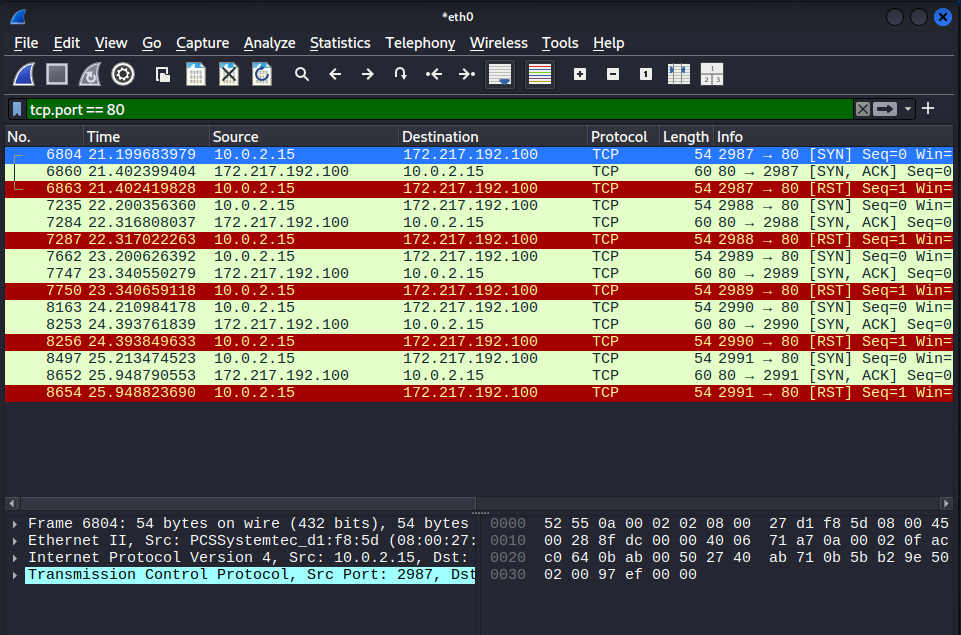


*Evidencia: resultados hping3 para puertos 22, 80, 443, 21*

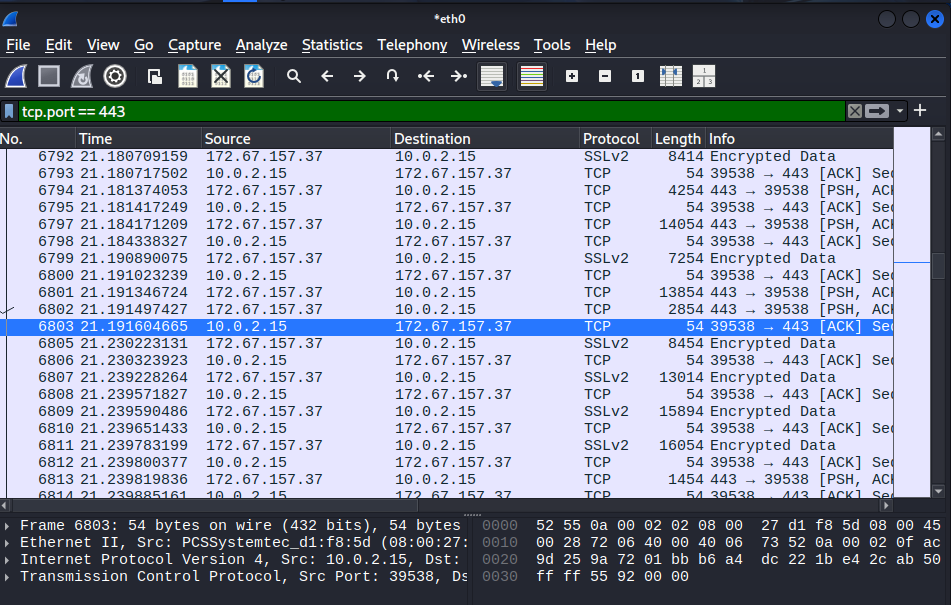
## 3.2 Evidencias en Wireshark



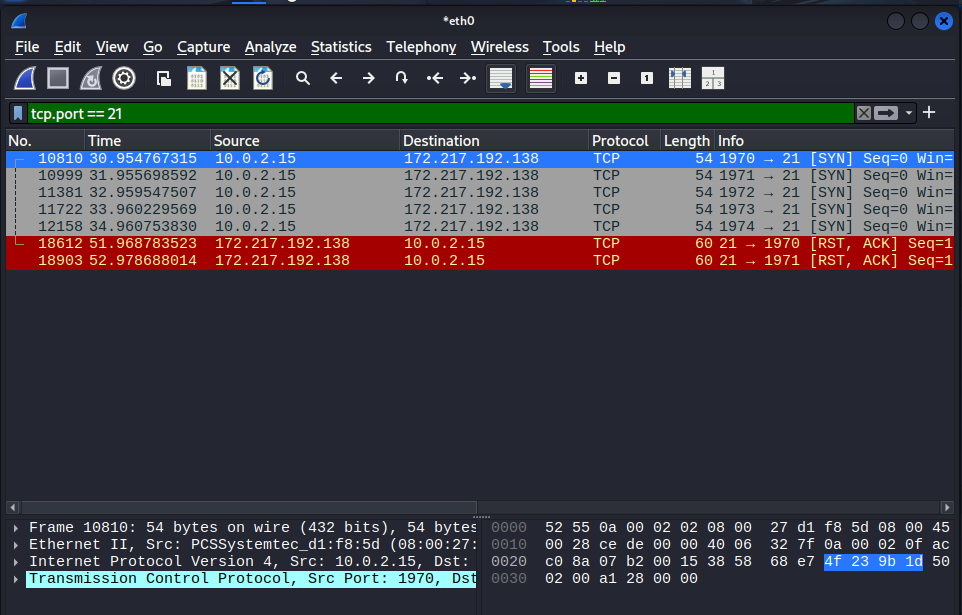
*Filtro tcp.port==22 (SSH) – respuesta RST/ACK (cerrado)*



*Filtro tcp.port==80 (HTTP) – respuesta SYN/ACK (abierto)*



*Filtro tcp.port==443 (HTTPS) – tráfico TLS/SSL cifrado*



*Filtro tcp.port==21 (FTP) – respuesta RST/ACK (cerrado)*

# Punto 4 – Conclusiones y Recomendaciones

## 4.1 Conclusiones

Del análisis realizado con las herramientas hping3 y Wireshark, se obtuvieron las siguientes conclusiones:

1. Con hping3 se comprobó la conectividad y estado de puertos comunes (22, 80, 443 y 21). Los resultados mostraron que los puertos 80 (HTTP) y 443 (HTTPS) se encuentran abiertos, mientras que los puertos 22 (SSH) y 21 (FTP) responden con RST/ACK, indicando que están cerrados. Esto refleja una política adecuada de exposición de servicios.

2. En las capturas de Wireshark se observó que el tráfico predominante corresponde a protocolos seguros como TLS/HTTPS (≈33%) y QUIC (≈6%). No obstante, aún se detectó un pequeño porcentaje de tráfico en HTTP sin cifrado, lo cual representa un riesgo potencial.

3. DNS funciona sobre UDP/53 sin cifrado, exponiendo consultas y respuestas a posibles ataques de spoofing o manipulación.

4. El protocolo ICMP respondió de manera correcta y útil para diagnóstico, aunque puede ser aprovechado para reconocimiento de red en entornos hostiles.

## 4.2 Recomendaciones

1. Migrar todos los servicios a HTTPS/TLS, evitando el uso de HTTP en texto plano para proteger la confidencialidad de los datos transmitidos.

2. Implementar DNS seguro mediante protocolos cifrados como DNS over HTTPS (DoH) o DNS over TLS (DoT) para proteger las consultas frente a intercepción o manipulación.

3. Restringir puertos innecesarios en los servidores públicos. Mantener cerrados puertos como FTP (21) y SSH (22) si no son requeridos, aplicando reglas de firewall más restrictivas.

4. Complementar las pruebas puntuales con sistemas de monitoreo continuo IDS/IPS (ej. Snort, Suricata, Wazuh) para detectar patrones anómalos y ataques en tiempo real.

5. Controlar el uso de ICMP, permitiéndolo únicamente en dispositivos necesarios para diagnóstico y aplicando limitación de tasas (rate limit) para mitigar intentos de abuso.