🛠️ Ejercicio Práctico

📌 **Título:** *Análisis del Flujo de Datos y Protocolos Activos en una Comunicación Web Segura*

🎯 Objetivo del ejercicio:

Simular, observar y analizar la interacción de varios protocolos del modelo TCP/IP durante una sesión de navegación web, utilizando herramientas de inspección de red como Wireshark o el navegador.

📝 Escenario:

Trabajas como técnico de soporte en una pequeña empresa. El gerente solicita que verifiques **qué protocolos se activan al acceder al portal interno seguro de la empresa:**
👉 `https://intranet.miempresa.local`

Debes simular esta navegación y **analizar paso a paso qué protocolos participan** desde que se ingresa la URL hasta que la página carga completamente.

🔍 Tu tarea:

1. **Analiza la ruta de datos en capas TCP/IP.**
   Completa el siguiente cuadro indicando qué protocolo participa en cada capa y qué rol cumple:

Capa TCP/IP	Protocolo(s) usado(s)	Función principal
Aplicación		
Transporte		
Internet		

Enlace de datos

---

2. **Realiza una captura de red local (opcional):**
    - Usa **Wireshark** o el inspector de red del navegador (`F12 > Red/Network`)
    - Accede a `https://intranet.miempresa.local`
    - Filtra los paquetes por protocolo (ej. TCP, DNS, TLS)
    - Toma nota de al menos 3 paquetes y su propósito

---

3. **Responde las siguientes preguntas breves:**

- ¿Qué protocolo tradujo el nombre de dominio en una IP?
- ¿Qué protocolo garantizó que los datos llegaron correctamente?
- ¿Qué protocolo cifró la conexión para que fuera segura?
- ¿Cuál fue el protocolo responsable del direccionamiento IP?

---

4. **Reflexiona:**

📌 ¿Por qué es importante que protocolos como HTTPS, TCP y DNS trabajen juntos?

✅ Resultado esperado:

- Reconocimiento de protocolos: **HTTPS, TCP, IP, DNS, Ethernet**
- Correcta asociación con las capas del modelo TCP/IP
- Comprensión básica de cómo se encapsulan y desencapsulan los datos
- Identificación de herramientas básicas de análisis de red

📄 Entrega sugerida:

- Cuadro de capas y protocolos completado
- Capturas de red (si se usó Wireshark o navegador)
- Respuestas breves a las preguntas
- Conclusión personal/reflexiva (5 líneas aprox.)

💡 Herramientas recomendadas:

- 🧰 **Wireshark** (captura real de tráfico)
- 🧪 **Firefox/Chrome DevTools**
- 🧱 **Cisco Packet Tracer** (en modo simulación) si no se permite usar redes reales

---

🛠️ Ejercicio Práctico – Solución

📌 *Análisis del Flujo de Datos y Protocolos Activos en una Comunicación Web Segura*

🎯 Objetivo

Simular y analizar los protocolos activos durante el acceso a un sitio web seguro dentro de una red local, identificando su relación con las capas del modelo TCP/IP y utilizando herramientas de análisis de red como Wireshark o las DevTools del navegador.

📝 Escenario

Durante una sesión de soporte, el gerente solicita un informe sobre los protocolos que se activan al visitar `https://intranet.miempresa.local`, un sitio interno de la empresa. El técnico debe inspeccionar el flujo de datos desde que se ingresa la URL hasta que la página carga completamente.

📑 Cuadro: Capas del Modelo TCP/IP y Protocolos Involucrados

Capa TCP/IP	Protocolo(s) usado(s)	Función principal
Aplicación	HTTPS, DNS	Permite la navegación web segura y la resolución de nombres de dominio a IP.
Transporte	TCP	Establece conexión confiable, garantiza la entrega de los datos y controla errores.
Internet	IP (IPv4/IPv6)	Direcciona y enruta los paquetes entre el cliente y el servidor web.
Enlace de Datos	Ethernet (o Wi-Fi)	Gestiona la transmisión física de los datos dentro de la red local.

📡 Captura de red (opcional)

Herramienta utilizada: DevTools de Chrome (F12 → Red)
Sitio accedido: `https://intranet.miempresa.local`
Protocolo observado en la columna "Protocol": HTTPS

Paquetes observados y su propósito:

1. **DNS:** Solicitud de resolución de `intranet.miempresa.local` → IP local
2. **TCP SYN:** Inicio de la conexión TCP con el servidor (puerto 443)
3. **TLS Handshake:** Establecimiento de la conexión segura entre cliente y servidor

(Opcional: insertar captura de pantalla del panel de red filtrado por "Protocol" o por "Name")

🧠 Preguntas clave

- **¿Qué protocolo tradujo el nombre de dominio en una IP?**
  → **DNS**
- **¿Qué protocolo garantizó que los datos llegaron correctamente?**
  → **TCP**

- **¿Qué protocolo cifró la conexión para que fuera segura?**
  → TLS (dentro de HTTPS)

- **¿Cuál fue el protocolo responsable del direccionamiento IP?**
  → IP (Internet Protocol)

🤔 Reflexión final

📌 *¿Por qué es importante que protocolos como HTTPS, TCP y DNS trabajen juntos?*

> Porque forman una cadena de funciones que permite que la comunicación sea segura, confiable y funcional. Si alguno falla, el usuario no podrá acceder al sitio correctamente. DNS localiza el servidor, TCP garantiza la entrega y orden de los datos, y HTTPS protege la privacidad e integridad de la información. Su trabajo conjunto es esencial para cualquier transacción web segura.

✅ Conclusión

Este ejercicio demuestra cómo múltiples protocolos interactúan en una simple acción como abrir una página web. El modelo TCP/IP permite entender esta interacción por capas, lo que resulta muy útil para análisis, solución de problemas y gestión de la seguridad en redes reales. Herramientas como Wireshark o las DevTools del navegador son clave para observar estos procesos en acción y verificar el buen funcionamiento de los servicios de red.

🧩 Documentación Técnica

Despliegue Local con Docker + Nginx + Dominio Personalizado

📌 Objetivo

Permitir acceder a una aplicación web en entorno local con una URL personalizada como:

http://intranet.miempresa.local

en lugar de `http://localhost`, usando **Docker**, **Nginx** y el archivo `hosts` de Windows.

🗂️ Estructura del Proyecto

```
mi-proyecto/
├── docker-compose.yml
├── Dockerfile
├── nginx.conf
├── app/
│   └── index.html
```

⚙️ Archivos del Proyecto

1. `docker-compose.yml`

```
version: '3.8'

services:
  nginx:
    image: nginx
    ports:
      - "80:80"
    volumes:
      - ./nginx.conf:/etc/nginx/conf.d/default.conf
      - ./app:/usr/share/nginx/html:ro
    networks:
      - mi_red

  app:
    build: .
    networks:
      - mi_red

networks:
  mi_red:
    driver: bridge
```

---

2. `Dockerfile`

```
FROM nginx:alpine
```

Este archivo es mínimo, puedes extenderlo más adelante para backend reales (Flask, FastAPI, Node, etc.)

---

3. `nginx.conf`

```
server {
    listen 80;
    server_name intranet.miempresa.local;

    root /usr/share/nginx/html;
    index index.html;

    location / {
        try_files $uri $uri/ =404;
    }
}
```

---

4. `app/index.html`

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Intranet de Mi Empresa</title>
</head>
<body>
  <h1>Bienvenido a la intranet de miempresa.local</h1>
</body>
</html>
```

🖥️ Configuración del archivo `hosts` en Windows

🔐 Requiere permisos de administrador

Pasos:

1. Abre **Bloc de notas como administrador**:
    - Busca “Bloc de notas” en el menú inicio
    - Clic derecho → **Ejecutar como administrador**

2. En Bloc de notas:
    - Archivo → Abrir
    - Ve a: `C:\Windows\System32\drivers\etc`
    - Cambia filtro: “*.txt” a “Todos los archivos”
    - Abre `hosts`

Agrega al final del archivo:

127.0.0.1 intranet.miempresa.local

3.
4. Guarda y cierra.

🚀 Ejecución del proyecto

Desde la raíz del proyecto:

docker-compose up --build

🌐 Acceso en navegador

Abre:
http://intranet.miempresa.local

🧪 Verificación rápida

Ping al dominio:

ping intranet.miempresa.local

Debe responder con: `127.0.0.1`.

✅ Resultado esperado

Visualizarás el contenido de `index.html` en tu navegador con la dirección:

http://intranet.miempresa.local

📌 Notas adicionales

- Si usas WSL2 o una red específica, asegúrate de que `Docker Desktop` expone correctamente los puertos.
- Puedes agregar certificados TLS locales (HTTPS) con `mkcert` si deseas navegación segura.
- Este entorno es ideal para desarrollo de **intranets**, **sistemas internos**, y pruebas locales con dominios simulados.

📋 Instrumento de Evaluación – Ejercicio Práctico

Análisis del Flujo de Datos y Protocolos en una Comunicación Web Segura
Puntaje total: 10 puntos
Nota mínima para aprobar: 6 puntos

Criterio Evaluado	Puntaje
Cuadro de capas TCP/IP completo con protocolos y funciones correctamente asignados	3 pts
Respuestas correctas a las preguntas clave (DNS, TCP, TLS, IP)	2 pts
Captura de red (Wireshark o DevTools) con al menos 3 paquetes comentados (DNS, TCP, TLS) *(opcional)*	1 pt

Explicación reflexiva sobre la importancia del trabajo conjunto de los protocolos	2 pts
Informe claro, bien estructurado y con evidencias (cuadro, respuestas, capturas si aplican)	2 pts

---