

Informe Experiencia 2

Benjamín Pavez Ortiz, 202173628-K

Angelo Russu Cisterna, 202173509-7

1 Respecto al laboratorio

1. ¿Cómo utilizó Wireshark para capturar y analizar las consultas y respuestas del servidor TCP y FTP? ¿Qué importancia tiene realizar este tipo de análisis? Explique a fondo y agregue capturas de pantalla de ser necesario.

Respuesta: Para capturar y analizar las consultas y respuestas, primero iniciamos Wireshark y comenzamos la captura de tráfico antes de establecer la conexión con el servidor TCP. Una vez iniciada la conexión, observamos las solicitudes y respuestas intercambiadas, las cuales quedaron registradas en Wireshark. Posteriormente, filtramos el tráfico para visualizar únicamente los paquetes TCP y FTP relevantes. Como resultado, pudimos identificar con precisión los paquetes de consulta y respuesta. La conexión establecida se muestra en la Figura 1.

Este tipo de análisis es crucial, ya que permite comprender el flujo de datos entre el cliente y el servidor, detectar posibles errores de comunicación y asegurar que los datos transmitidos sean correctos.

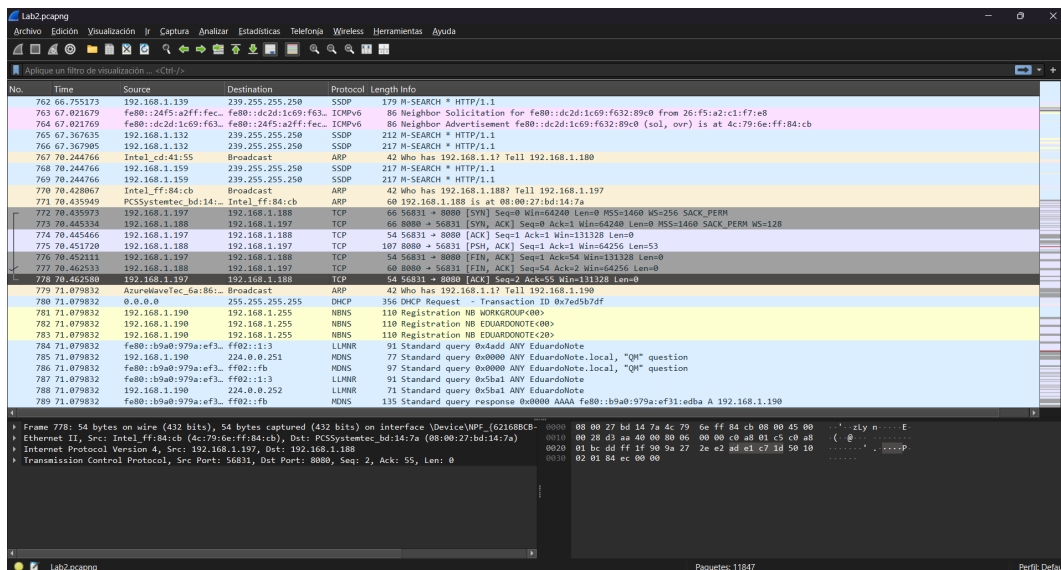
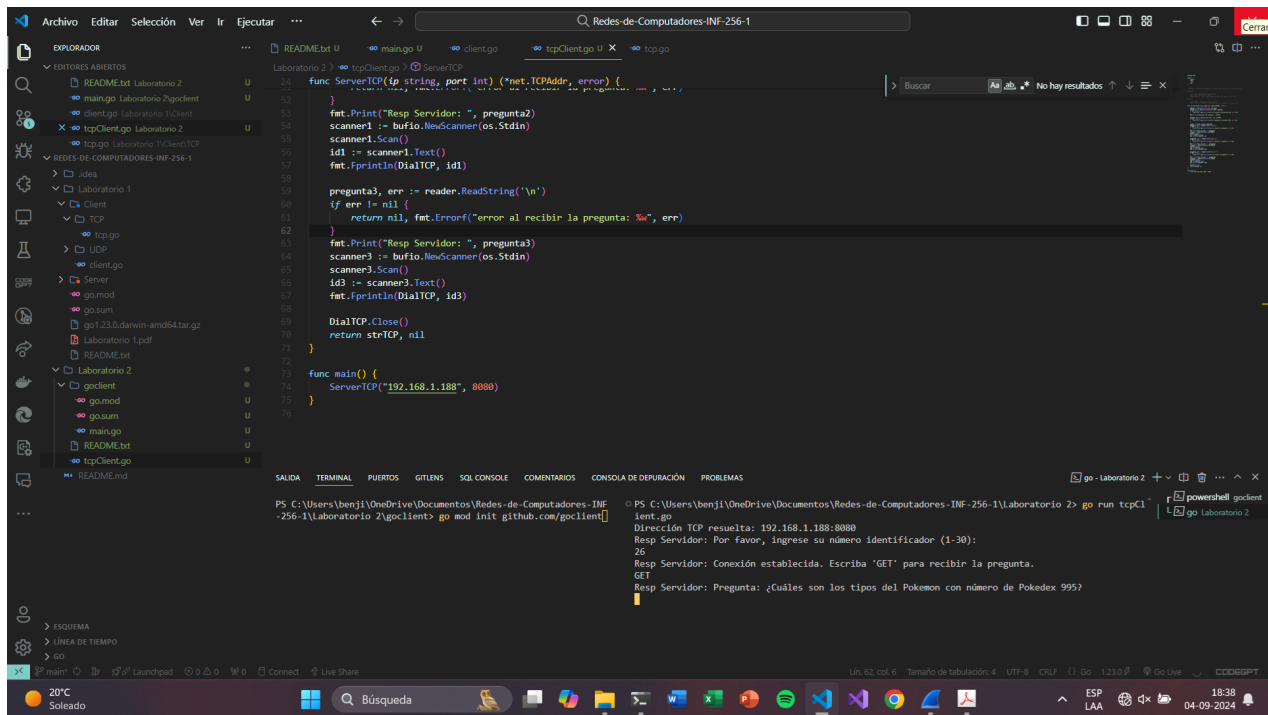


Figure 1: En la figura se muestra una conexión TCP con destino a la IP 192.168.1.188, que corresponde a la del laboratorio.

2. ¿Cuál fue su pregunta única y su respuesta correspondiente? ¿Qué credenciales utilizó para ingresar al servidor FTP? ¿Qué dirección IP se le asignó durante la experiencia? Incluya capturas de pantalla de ser necesario.

Respuesta: La pregunta enviada por el Servidor fue: *¿Cuáles son los tipos del Pokemon con el número de Pokedex 995?* y cuya respuesta es *Roca Electrico*. Las credenciales utilizadas para acceder al servidor FTP fueron: Usuario: **grupo26** y Contraseña: **RocaElectrico**. Además, la dirección IP que se nos asignó en el laboratorio fue: **192.168.1.188**.



The screenshot shows a Go IDE with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The code editor displays a Go program for a TCP server. The terminal shows the execution of the program, which listens on IP 192.168.1.188 and port 8080. It receives a connection from 192.168.1.188:8080 and prints the question: "¿Cuáles son los tipos del Pokemon con número de Pokedex 995?".

```
func ServerTCP(ip string, port int) (*net.TCPAddr, error) {  
    scanner1 := bufio.NewScanner(os.Stdin)  
    scanner1.Scan()  
    id1 := scanner1.Text()  
    fmt.Fprintln(DialTCP, id1)  
  
    pregunta3, err := reader.ReadString('\n')  
    if err != nil {  
        return nil, fmt.Errorf("error al recibir la pregunta: %v", err)  
    }  
  
    fmt.Print("Resp Servidor: ", pregunta3)  
    scanner3 := bufio.NewScanner(os.Stdin)  
    scanner3.Scan()  
    id3 := scanner3.Text()  
    fmt.Fprintln(DialTCP, id3)  
  
    DialTCP.Close()  
    return strTCP, nil  
}  
  
func main() {  
    ServerTCP("192.168.1.188", 8080)  
}
```

Terminal output:

```
PS C:\Users\benji\OneDrive\Documents\Redes-de-Computadores-INF-256-1\Laboratorio 2\goclient> go mod init github.com/goclient  
go: creating new go.mod: module github.com/goclient  
go: to add module requirements and sums:  
  go get  
PS C:\Users\benji\OneDrive\Documents\Redes-de-Computadores-INF-256-1\Laboratorio 2> go run tcpClient.go  
Dirección TCP resulta: 192.168.1.188:8080  
Resp Servidor: Por favor, ingrese su número identificador (1-30):  
26  
Resp Servidor: Conexión establecida. Escriba 'GET' para recibir la pregunta.  
GET  
Resp Servidor: Pregunta: ¿Cuáles son los tipos del Pokemon con número de Pokedex 995?
```

Figure 2: Captura de pantalla del programa y la terminal que muestra la pregunta

3. ¿Cómo fue la interacción que realizó con el servidor TCP y FTP? Explique a través de un diagrama.

Respuesta:

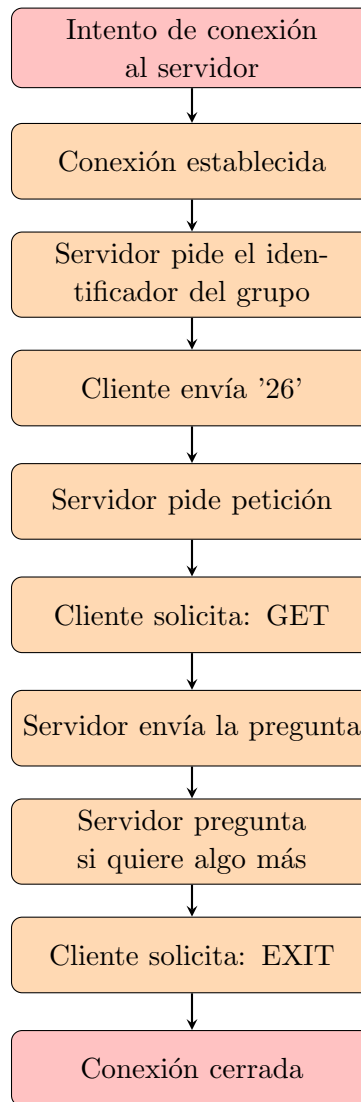


Figure 3: Interacción con el Servidor TCP

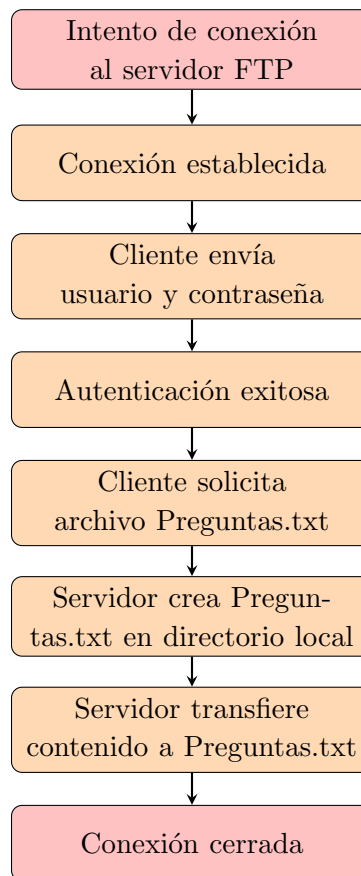


Figure 4: Interacción con el Servidor FTP

2 Respecto al protocolo FTP

1. ¿Cuáles son los principales problemas de seguridad del protocolo FTP?

Respuesta: El protocolo FTP presenta varias vulnerabilidades. Principalmente, transmite datos y credenciales sin encriptar, lo que facilita que los atacantes intercepten y lean la información. Además, FTP carece de mecanismos de autenticación de datos, permitiendo que la información pueda ser alterada sin que el destinatario pueda verificar su integridad. Para mejorar la seguridad, se recomienda el uso de alternativas como FTPS o SFTP, que ofrecen cifrado y autenticación robusta.

2. ¿Cuáles son los permisos que el servidor puede darle al usuario? ¿Qué tipos de problemas puede conllevar esto?

Respuesta: Los permisos que un servidor FTP puede otorgar a un usuario suelen incluir:

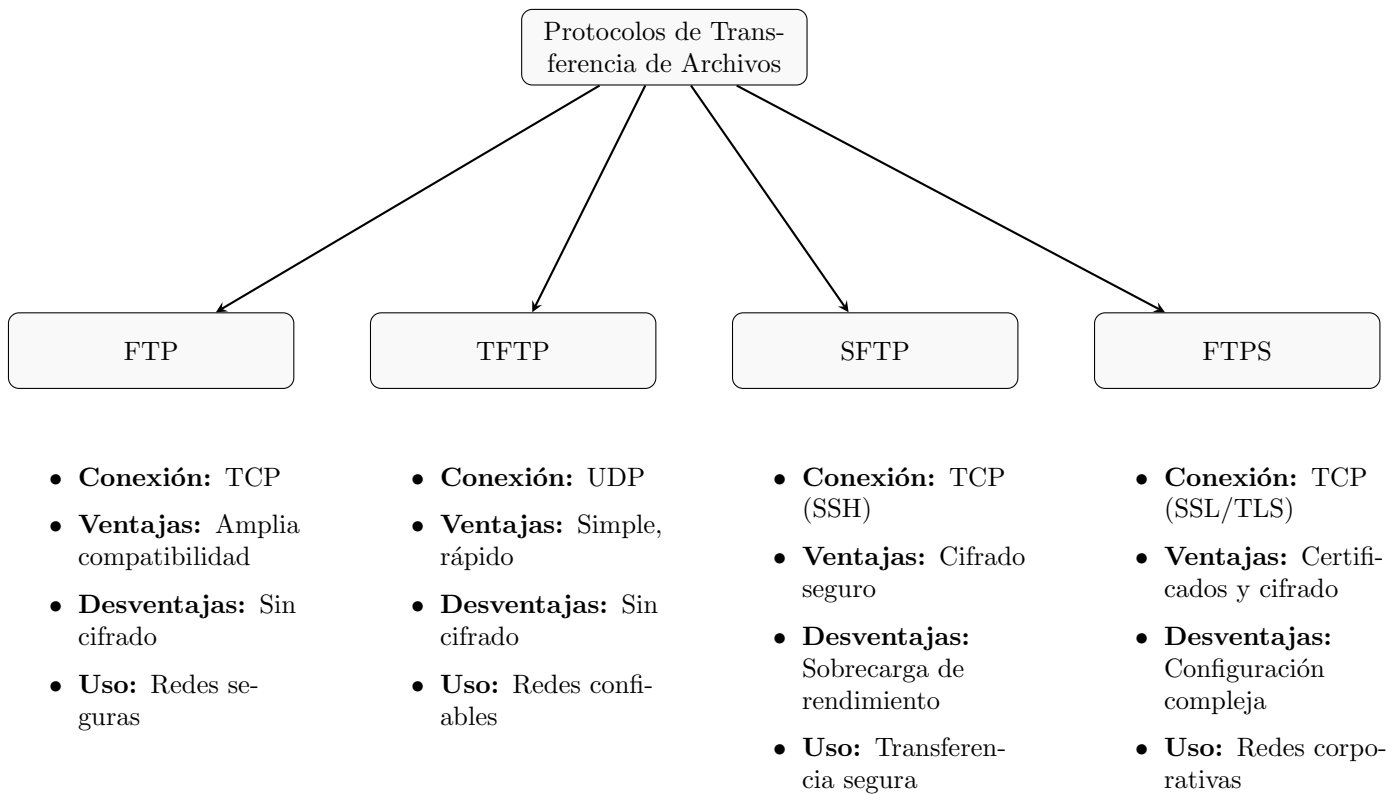
- **Lectura:** El usuario puede ver y descargar archivos.
- **Escritura:** El usuario puede subir archivos.
- **Modificación:** El usuario puede modificar archivos existentes.

- **Eliminación:** El usuario puede borrar archivos o directorios.

Problemas potenciales:

- **Alteración de archivos:** Los usuarios pueden modificar archivos relevantes.
 - **Seguridad:** Archivos sin cifrar pueden ser manipulados.
3. Realice un diagrama entre FTP, TFTP, SFTP, FTPS, explicando diferencias, ventajas/desventajas, algunos casos de uso y tipo de conexión (TCP o UDP).

Respuesta:



3 Conclusiones

1. ¿Que conclusiones puede extraer de la experiencia? ¿Que usos potenciales tienen los servidores FTP? ¿utilizarían este protocolo o buscarían alternativas?

Respuesta: El protocolo FTP es útil en redes seguras debido a su facilidad de implementación y rápida transferencia de archivos. Sin embargo, para aplicaciones donde la seguridad es una prioridad, es preferible utilizar FTPS o SFTP por sus características de cifrado y autenticación.