EJERCICIO 1

TELNET

1. ¿Cuál es la dirección IP del cliente y cuál es la del servidor?

Cliente: 192.168.12.1 y Servidor: 192.168.12.2

2. ¿Qué credenciales se han utilizado para acceder al servidor?

Se ha usado una contraseña para acceder: cisco

3. ¿Qué tipo de sistema es el servidor?

Parece un router ya que dice que es cisco

4. ¿Qué comando(s) ha ejecutado el cliente en el servidor?

Solo ha ejecutado uno, exit

FTP

1. ¿Cuál es la dirección IP del cliente y cuál es la del servidor?

Cliente: 192.168.1.182 y Servidor: 192.168.1.231

2. ¿Qué credenciales se han utilizado para acceder al servidor?

USER: ftp y PASS ftp

3. ¿Qué tipo de sistema es el servidor?

215 UNIX Type: L8

4. ¿Qué comando(s) ha ejecutado el cliente en el servidor?

- USER ftp: Envía el nombre de usuario.
- PASS ftp: Envía la contraseña.
- SYST: Solicita información sobre el sistema del servidor.
- FEAT: Lista las características soportadas por el servidor FTP.
- PWD: Muestra el directorio actual.
- EPSV: Entra en modo pasivo extendido.
- LIST: Lista los archivos en el directorio actual.
- TYPE I: Cambia al modo binario para transferencias.
- SIZE resume.doc: Obtiene el tamaño del archivo resume.doc.
- RETR resume.doc: Descarga el archivo resume.doc.
- MDTM resume.doc: Obtiene la fecha y hora de modificación del archivo resume.doc.
- CWD uploads: Cambia al directorio uploads.
- PWD: Muestra el directorio actual (en este caso, /uploads).
- STOR README: Sube un archivo llamado README.
- MKD testdir: Intenta crear el directorio testdir (fallido porque ya existe).
- MKD testerdir: Crea un directorio llamado testerdir.
- RMD testerdir: Elimina el directorio testerdir.

- CWD ..: Vuelve al directorio anterior.
- PWD: Muestra el directorio actual.
- TYPE A: Cambia al modo ASCII.
- LIST: Lista los archivos en el directorio actual nuevamente.
- SITE CHMOD 777 resume.doc: Intenta cambiar los permisos del archivo resume.doc (fallido).
- QUIT: Termina la sesión.

EJERCICIO 2

Ejercicio 2.1 (TLS 1.3)

1. ¿Cuándo (de qué trama a qué trama) se procede con el proceso de handshake

(sesión SSL), tal y como se ha explicado en teoría?

A partir del Client Hello (trama 4), el Server Hello (trama 6) y Change Cipher Spec (trama 8)

2. En esta conexión se utiliza TLS1.3. ¿Dónde se negocia exactamente la versión de

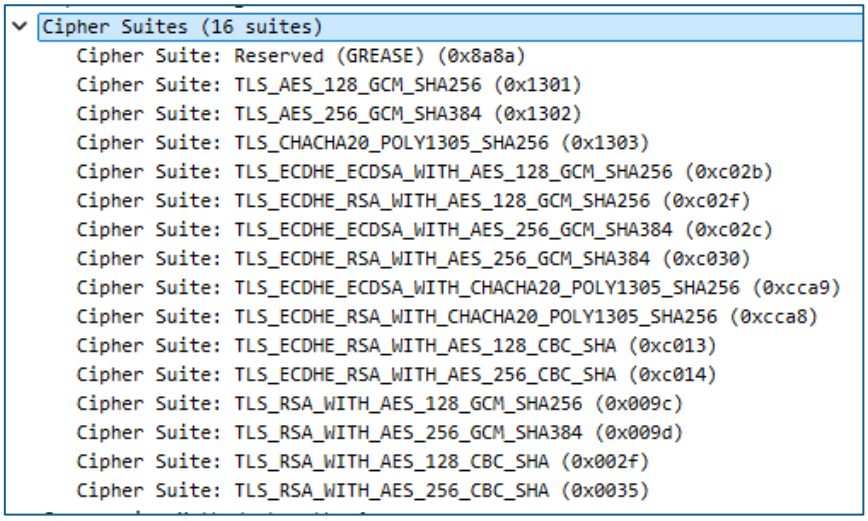
TLS que se utiliza?

El cliente en Client Hello le muestra al servidor as versiones que soporta y éste selecciona una de ellas en la trama Server Hello

3. En la parte del cliente, ¿en qué trama se puede ver las suites de cifrado que

soporta el cliente? ¿Cuáles son?

En Client Hello:



4. ¿Qué suite de cifrado se acepta finalmente para el proceso de conexión?

La que aparece en Server Hello: TLS_AES_256_GCM_SHA384 (0x1302)

5. En TLS1.3, no es posible ver la trama en la que se envía el certificado digital del

servidor. ¿Por qué ocurre eso?

• Adicionalmente, de forma opcional: ¿Sería posible inferir cuál es la trama en la que el servidor envía al cliente su certificado?

En TLS1.3, no se envía el certificado digital del servidor en un mensaje explícito dentro del handshake. En cambio, TLS1.3 usa un mecanismo llamado "Authenticated Encryption with Associated Data" (AEAD) en el que el certificado y otros elementos relacionados se integran directamente en los mensajes del protocolo, como en la trama 6. Por lo tanto, no hay una trama separada en la que se envíe solo el certificado.

Aunque no hay una trama explícita para el certificado, se puede inferir que el servidor está autenticado ya que lo hemos incluido en nuestro código y estará en alguna parte del Server hello.

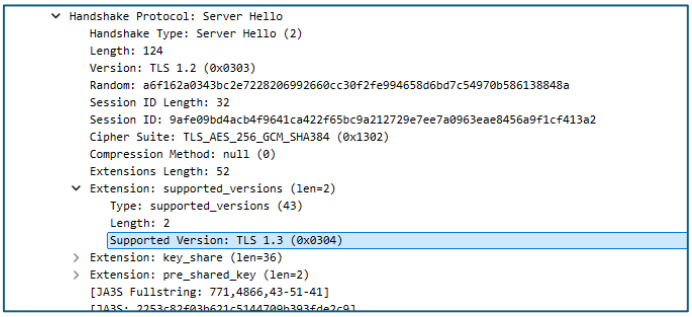
Ejercicio 2.2 (TLS 1.2)

1. ¿Cuándo (de qué trama a qué trama) se procede con el proceso de handshake (sesión SSL), tal y como se ha explicado en teoría?

Se inicia con Client Hello en la trama 4, sigue ServerHello en la trama 6 y Change Cipher Spec en la 8

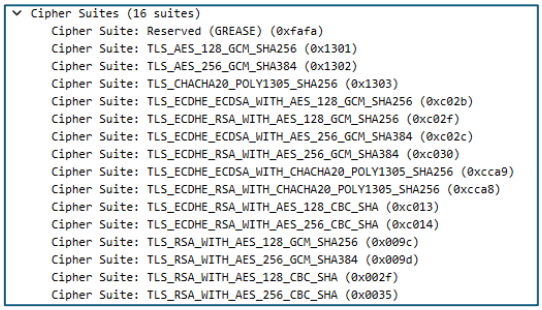
2. En esta conexión se utiliza TLS1.2. ¿Dónde se negocia exactamente la versión de TLS que se utiliza?

En el Client Hello se dice que se soporta hasta 1.3 y aunque el servidor también soporta 1.3 selecciona el 1.2 como le hemos indicado:



3. En la parte del cliente, ¿en qué trama se puede ver las suites de cifrado que soporta el cliente? ¿Cuáles son?

Igual que en 1.3 se ve en Client Hello



4. ¿Qué suite de cifrado se acepta finalmente para el proceso de conexión?

El servidor selecciona TLS_AES_256_GCM_SHA384 (0x1302)

5. ¿En qué trama se envía el certificado digital del servidor? En esa trama, ¿Dónde se encuentra vuestro nombre (el "common name" cuando creasteis el certificado)? ¿Cuál es la clave pública del servidor?

Todo esto aparece en la trama 6. Después del Server Hello, está otra parte del Handshake Protocol que es Certificate. Aquí se envía el certificado digital del servidor. Se puede ver el common name en varios campos. Dentro de Certificates –→ Certificate –→ signedCertificate –→ issuer o en subject. Por último, la clave pública del servidor aparece justo después en subjectPublicKeyInfo.

6. ¿El servidor se autentica al cliente? ¿Y el cliente al servidor?

El servidor sí se autentica como hemos visto en la pregunta anterior pero en cliente no, ya que no le envía el certificado.