Wireshark, encapsulamiento y ARP

1 Introducción

En esta primera Prueba de Evaluación Continua vamos a realizar una práctica utilizando **Wireshark** que es una aplicación orientada al análisis de redes y protocolos. Wireshark es la aplicación más universal en su ámbito y permite capturar los paquetes enviados y/o recibidos por una determinada interfaz de red del ordenador en el que se instala. Es multiplataforma y desde su página oficial[1] se puede descargar la versión para Windows o para Mac. La versión para Linux puede instalarse directamente a través del gestor de paquetes de la mayoría de distribuciones.

La toma de contacto con una herramienta de análisis de redes como Wireshark es uno de los objetivos de esta práctica. Además, el uso de esta aplicación nos va a permitir experimentar también con algunos conceptos estudiados en la asignatura como el encapsulamiento o el estudio y análisis de protocolos de red. En este caso vamos a trabajar con el protocolo ARP que permite obtener la dirección física de capa de enlace a partir de la dirección lógica de la capa de red.

También vamos a comprobar que la captura de paquetes, con una herramienta de análisis como Wireshark, nos revela la necesidad de realizar todas las comunicaciones de manera cifrada, ya que en una comunicación no cifrada, este tipo de herramientas nos permiten visualizar todo el contenido que se propaga a través de la red. Basta con tener acceso físico a la misma.

2 Instrucciones

Este documento le va a ir guiando en la realización de la actividad práctica y conforme vaya avanzando en la ejecución le realizará una serie de preguntas que debe responder rellenando las casillas correspondientes de este mismo documento. Los espacios de respuesta están establecidos con mucha holgura para que se pueda responder de manera completa y con precisión. Debe ser preciso a la vez que conciso puesto que el texto que sobrepase el cuadro asignado podría no tenerse en cuenta.

Junto con este documento que contiene el enunciado ha debido obtener dos ficheros con sendas capturas de Wireshark: **Captura_FTP_plano.pcapng** y **Captura_FTP_cifrado.pcapng**.

2.1 Equipamiento necesario

Para la realización de esta práctica basta con cualquier ordenador conectado a Internet. Esto supone la utilización de un rúter que proporcione ese acceso. En este ordenador deberemos instalar la aplicación Wireshark que se puede descargar gratuitamente de su propio portal. Se recomienda una conexión cableada y evitar las conexiones WiFi.

En la segunda parte será necesario el uso de otro dispositivo en la misma red. Puede ser otro ordenador, una smartTV, impresora, tablet, teléfono móvil, etc.

2.2 Documentación a entregar

Cuando haya finalizado la actividad deberá subir al curso virtual, dentro del plazo establecido, un archivo zip cuyo nombre será **Nombre_Apellido1_Apellido2.zip** y que contenga los siguientes ficheros:

1 Sitio oficial de Wireshark: https://www.wireshark.org

- Este mismo documento convenientemente relleno y cambiando el nombre actual por su nombre y apellidos: **Nombre_Apellido1_Apellido2.pdf**. Después de guardar el archivo asegúrese que todas las casillas tienen la información que usted haya introducido.
- Un archivo de captura, en formato Wireshark, solicitado más adelante. Utilice el mismo esquema base para el nombre de este archivo: **Nombre_Apellido1_Apellido2.pcapng**. Asegúrese que este archivo que adjunta se corresponde con el utilizado para responder las preguntas. De lo contrario, las respuestas podrían ser incorrectas y por lo tanto no puntuarían. Para asegurar esto conviene que haga la captura, la guarde en disco y después trabaje sobre el archivo guardado en lugar de hacerlo al revés.

2.3 Puntuaciones

A lo largo de la realización de la práctica se le irán planteando una serie de preguntas con varios apartados o subpreguntas en cada una. En la cabecera de cada pregunta figura la contribución de esa pregunta en la valoración final. Por simplicidad, estas valoraciones aparecen siempre como un número entero positivo, por lo que la suma total supera el valor 10 estándar. La nota final se reescalará sobre 10. Es decir:

$$Nota\,final = Puntos\,obtenidos \times \frac{10}{Total\ puntos\ posibles}$$

Cada fallo en la respuesta de un apartado de cada pregunta supone una penalización de 1 punto en esa pregunta pero en ningún caso se podrán obtener puntuaciones negativas. Por ejemplo, una pregunta con 5 apartados y ponderada con 3 puntos, se calificará de la siguiente forma:

- 0 fallos: 3 puntos
- 1 fallo: 2 puntos
- 2 fallos: 1 punto
- 3 o más fallos: 0 puntos

En los cuadros de respuesta libre, no basta con responder el valor exacto, si se pide, sino que deberá incluirse la justificación razonada de la respuesta. Si la explicación es errónea, incompleta o imprecisa se considerará nula.

Los apartados con texto atenuado, son únicamente como referencia para otras preguntas y no tienen ningún peso en la calificación. No obstante deben rellenarse convenientemente porque de lo contrario las respuestas que se refieran a esta información no serán válidas.

3 Entorno de Wireshark

El manejo de Wireshark es muy intuitivo y existe abundante documentación tanto en el propio portal de Wireshark como en otros muchos portales de Internet, foros y libros específicos. Aquí, simplemente, vamos a realizar una pequeña introducción para desenvolverse con la aplicación desde el primer instante.

La instalación no presenta ninguna particularidad reseñable, basta con aceptar las opciones por defecto. Al iniciar la aplicación nos aparecerá una primera pantalla (Figura 1) en la que, la principal acción, es elegir la interfaz de red en la que queremos capturar los paquetes. Junto al nombre de cada interfaz Wireshark muestra una pequeña traza que representa la actividad de esa interfaz. Esto nos permite conocer las interfaces activas y nos ayuda a elegir la correcta.

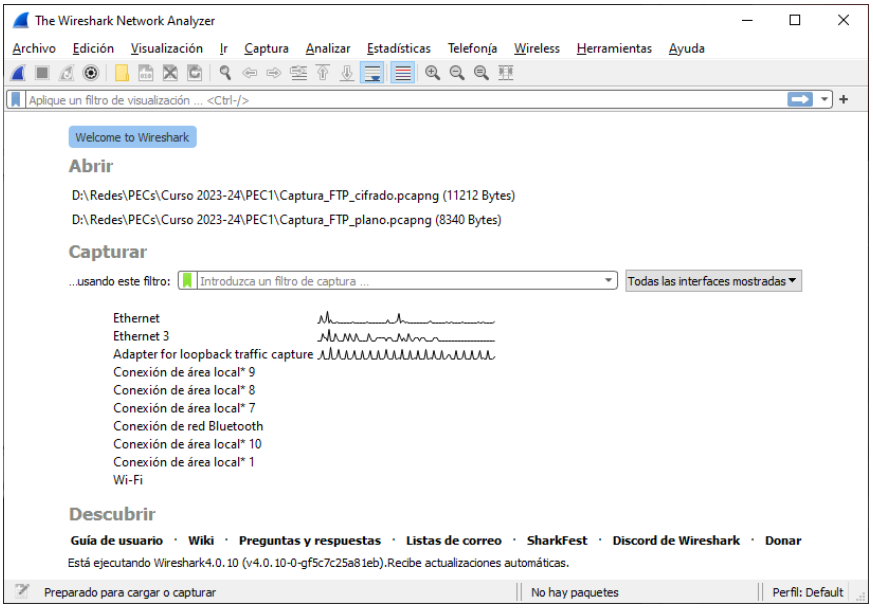

Figura 1 Pantalla de inicio de Wireshark donde se puede elegir la interfaz de captura

Para continuar, basta con hacer doble clic sobre el nombre de la interfaz en la que queramos capturar paquetes. Las capturas solo se realizarán sobre la interfaz de red especificada en este paso por lo que debe estar seguro de que selecciona la interfaz correcta si su ordenador dispone de más de una.

Aquí también podemos especificar un filtro de captura para los paquetes. De momento no haremos uso de este tipo de filtrado y capturaremos todos los paquetes. Posteriormente podremos filtrar los paquetes que se visualizan. Ese posterior filtrado de visualización nos permitirá despejar la lista de capturas y acceder mucho más fácilmente a un determinado tipo de paquetes o a paquetes que cumplan ciertas condiciones. Esta pantalla también nos permite recuperar las últimas capturas guardadas en el disco o acceder a la guía de uso de la aplicación, disponible en Internet, haciendo clic sobre el enlace etiquetado como **Guía de usuario** en la parte inferior de esa pantalla.

Una vez que hayamos pasado esa primera pantalla, Wireshark comenzará, inmediatamente, a capturar paquetes en la interfaz seleccionada y pasará a mostrarnos la ventana principal de la aplicación (Figura 2). Para detener esa captura basta con pulsar sobre el botón que muestra el icono de un cuadrado rojo.

Desde esta pantalla principal tenemos acceso a toda la información necesaria para la realización de la práctica. En la parte superior tenemos el típico menú principal y justo debajo una barra de herramientas con los botones que dan acceso a las acciones más frecuentes. Si el aspecto de la pantalla es distinto al mostrado en la Figura 2, puede cambiarlo desde el menú **Edición/Preferencias/Apariencia/Diseño**.

Debajo de esta barra de herramientas podemos ver un cuadro de edición donde podremos establecer condiciones de filtrado. En esta práctica haremos un uso simple de este filtrado. Mediante el botón de la izquierda de este cuadro de edición se puede acceder a algunos filtros preconfigurados y a la definición de otros nuevos que podemos crear y aparecerán posteriormente en el listado.

A continuación podemos ver una tabla con filas coloreadas. Aquí es donde se mostrarán los detalles fundamentales de todos los paquetes capturados. Cada línea de esta tabla se corresponde con un paquete. El color indica el tipo de protocolo para poder identificarlos de una manera rápida y muy visual. El contenido de cada fila dependerá del tipo de protocolo que utiliza el paquete capturado en el

nivel más alto de encapsulamiento. La aplicación viene configurada por defecto con una serie de columnas que se corresponden con los parámetros más relevantes y que son comunes a todos los tipos de paquetes que podemos encontrar. Podemos reordenar la posición de las columnas simplemente pinchando sobre la etiqueta de la cabecera y arrastrando hacia la posición deseada. También podemos añadir nuevas columnas o eliminar alguna de las que se muestran haciendo clic derecho sobre la línea de cabecera y eligiendo la opción **Preferencias de columna...** Esto nos abrirá un cuadro de diálogo que en la parte inferior tiene dos botones para añadir y eliminar columnas. Una vez añadida una nueva columna podremos cambiar el nombre que se visualiza en la cabecera de la tabla y también el tipo de información que se mostrará en la columna. Para estos cambios basta con hacer doble clic sobre el elemento que queramos modificar.

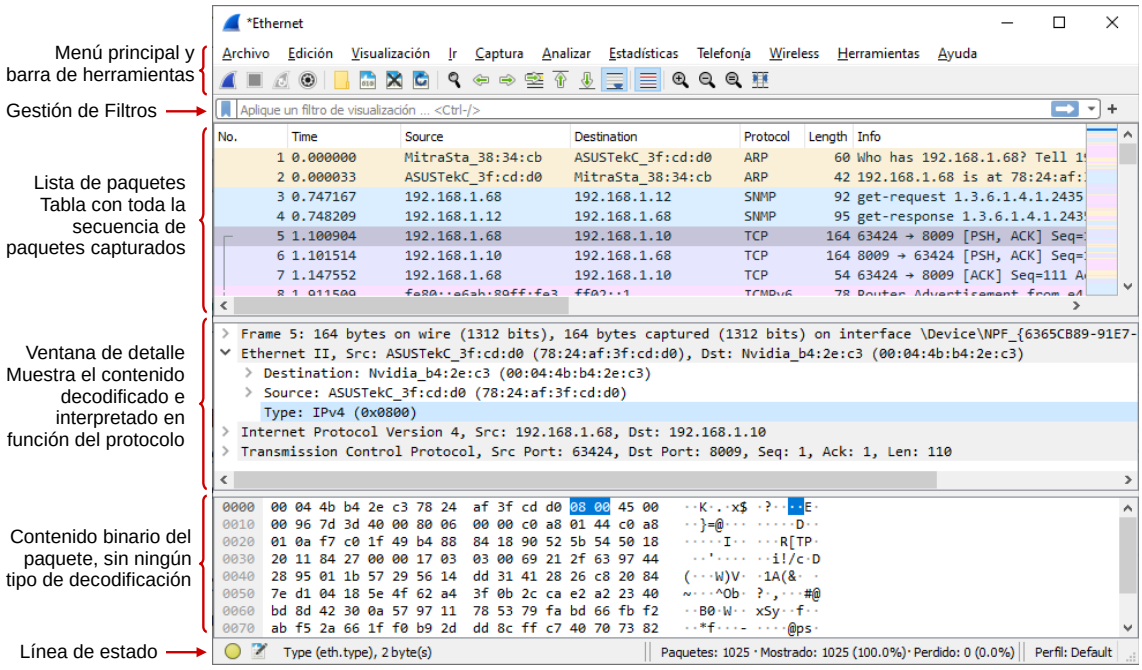

Figura 2 Pantalla principal de la aplicación Wireshark donde se muestran los elementos principales

En la tabla coloreada de la Figura 2 podemos hacer clic sobre cualquiera de sus filas para seleccionar el paquete correspondiente. Al hacerlo, el contenido del mismo aparecerá en las dos ventanas inferiores: en el centro de la pantalla veremos la ventana de detalles que comentaremos luego y en la parte inferior la ventana con el contenido binario del paquete seleccionado. En esta ventana inferior se muestra el contenido del paquete en crudo, sin ningún tipo de decodificación ni interpretación de la información contenida en cada paquete. Aquí tenemos tres columnas. En la parte izquierda tenemos una columna con la posición del primer byte de cada fila de la columna central y nos permite conocer, por tanto, la posición exacta de cada byte dentro del paquete completo. En la del centro vemos una secuencia de cifras hexadecimales que representan el contenido binario del paquete seleccionado. En la parte derecha se muestra la misma información pero de manera alfanumérica. Esto puede resultar interesante en los protocolos que trabajen a nivel de texto. Para el resto, esta tercera columna simplemente mostrará caracteres aparentemente sin sentido.

Hemos dejado para el final la ventana más relevante de Wireshark porque es en la que podemos ver el contenido de los distintos paquetes pero decodificados e interpretados por la propia aplicación. Aquí se mostrará en cada momento el contenido del paquete que esté seleccionado en la tabla superior. Wireshark reconoce el protocolo utilizado en cada nivel del modelo de capas y muestra su contenido de una manera amigable. Realmente es la misma información que podemos ver en binario en la

ventana inferior pero aquí ya la tenemos identificada, decodificada y correctamente interpretada por la aplicación.

En esta ventana podemos ver varías líneas que comienzan por un símbolo **>**. Cada una de estas líneas representa la información de cada uno de los protocolos tal y como van encapsulados en los paquetes de las distintas capas. La primera de estas líneas es un poco especial porque no muestra el contenido de ningún paquete sino los datos relativos al paquete en bruto capturado, tal como su número de orden, la hora en que fue capturado, su tamaño, etc.

El desencapsulado de datos comienza con la segunda línea donde se muestra la trama de capa de enlace. La siguiente mostrará el datagrama de capa de red, la siguiente la información de capa de transporte, si la hubiese, y conforme vayamos avanzando en estas líneas iremos viendo la información de la siguiente capa encapsulada en la anterior. No todas los paquetes mostrarán aquí el mismo número de líneas puesto que no todos los paquetes encapsulan contenido de otras capas superiores.

Al hacer clic sobre el símbolo **>** de cualquiera de estas líneas se despliega el contenido interpretado de esa capa donde se pueden ver los distintos parámetros de la cabecera del protocolo correspondiente. Si hacemos clic sobre cualquiera de estos parámetros de cabecera, su contenido binario/hexadecimal se coloreará en la ventana inferior. Esto también funciona a la inversa: al hacer clic sobre cualquiera de los bytes de la ventana inferior, se desplegará automáticamente la línea que muestra el parámetro al que pertenece el byte seleccionado. Al desplegar las líneas de la ventana central, solo se muestran los elementos de la cabecera correspondiente puesto que los datos de esa capa se muestran en la línea siguiente ya que corresponderán a un nivel de capa superior.

Para iniciar una captura basta con pulsar sobre el primer icono de la barra de herramientas (con forma de aleta de tiburón, que es el logo de la aplicación). Al hacerlo, nos preguntará si deseamos guardar en un archivo la captura actual. Para detener una captura debe pulsarse el botón con un cuadrado rojo. Conviene detener la captura lo antes posible para reducir el número de paquetes capturados y simplificar posteriormente el recorrido por los múltiples paquetes.

Una vez introducido el entorno de trabajo vamos a utilizarlo para estudiar el encapsulamiento de paquetes y el protocolo ARP.

4 Parte 1: Encapsulamiento, direccionamiento y seguridad

En esta primera parte vamos a analizar dos archivos de captura correspondientes a sendas conexiones utilizando el protocolo FTP (File Transfer Protocol). FTP es uno de los protocolos estándar de capa de aplicación para la transferencia eficiente de archivos entre un servidor y un cliente. Estos dos archivos los habrá obtenido junto con este documento. La diferencia entre estos dos archivos de captura es que en un caso la conexión se ha realizado mediante el protocolo FTP ordinario (archivo **Captura_FTP_plano.pcapng**), que trabaja sin ningún tipo de cifrado y en el otro utilizando una conexión cifrada (archivo **Captura_FTP_cifrado.pcapng**). Para la realización de la práctica no es necesario conocer el detalle de este protocolo que se estudiará muy brevemente en el tema correspondiente a la capa de aplicación. Para generar estos archivos de captura se ha utilizado un cliente y un servidor FTP, concretamente los de Filezilla que están disponibles gratuitamente en https://filezilla-project.org/

Cargue en Wireshark el fichero correspondiente a la captura de paquetes de la conexión sin cifrar (menú **Archivo/Abrir** o **CTRL+0**). Localice los paquetes correspondientes a la comunicación FTP y responda a las siguientes preguntas.

Pregunta 1 (4 puntos)

Wireshark es una aplicación de tipo *sniffer* que captura la información que pasa a través de una determinada interfaz de red.

¿Qué tipo de paquetes son los que captura Wireshark?

- (•) Tramas de capa de enlace
- ( ) Datagramas de red
- ( ) Segmentos de capa de transporte
- ( ) Datagramas de usuario
- ( ) Mensajes de aplicación
- ( ) Otro tipo de paquetes
- ( ) No sabe / No contesta

Localice el primer paquete específico del protocolo FTP. ¿Qué número de orden tiene dentro de toda la secuencia de captura?

Número del primer paquete FTP: 31

Ahora vamos a inspeccionar el encapsulmiento en las distintas capas. Respecto a ese primer paquete FTP ¿Qué tipo de trama/protocolo se utiliza en cada una de las capas y cual es la longitud de la cabecera y de la parte de datos en cada caso? Exprese las longitudes en bytes.

	Tipo/nombre de paquete y protocolo	Longitud de la cabecera	Longitud de los datos
Capa de enlace	Ethernet II (Ethernet)	14	91
Capa de red	Internet Protocol Version 4 (IPV4)	20	71
Capa de transporte	Transmission Control Protocol (TCP)	20	51
Capa de aplicación	File Transfer Protocol (FTP)		51

Pregunta 2 (3 puntos)

Identifique la máquina que presta el servicio FTP (servidor) y la máquina que lo solicita (cliente) e indique, las direcciones MAC (en hexadecimal) e IP de ambas máquinas. Averigüe el número de puerto que se está utilizando en cada extremo de la conexión:

	Cliente	Servidor
Direcciones MAC	00:04:0f:04:c4:3a	00:13:a9:2c:b0:e4
Direcciones IP	108.157.10.1	108.108.90.10
Número puerto	35526	21

Describa brevemente cómo ha identificado a cada una de estas máquinas y cómo ha determinado cual actúa de cliente y cual de servidor.

Hay dos máquinas, una es la de origen (source) y la otra destination (destinatario). La máquina origen tiene unos datos determinados que se encuentran en las diversas capas, dentro de la trama/protocolo. En la destinatario la identificamos de la misma manera que la de origen, pero con sus respectivos datos. Por las diversas direcciones no podemos determinar cuál es cliente y servidor. E intentado buscar las MAC en internet para ver a qué dispositivos corresponden, pero no encontré nada. Lo identifico por la conversación. El cliente es el que inicia la comunicación y el servidor responde con una response(respuesta) que es lo que vemos en el FTP. La respuesta del servidor es ese mensaje en el que da la bienvenida a la PEC 1, que lo envía el source.

Pregunta 3 (3 puntos)

Volviendo al primer paquete del protocolo FTP:

¿Quien envía este primer paquete FTP ? ○ Cliente ◉ Servidor ○ En blanco

¿Le parece razonable o intuitiva esta respuesta? Indique el motivo. (El motivo de por qué le parece (o no) intuitiva la respuesta, no el por qué es así que se verá en temas posteriores).

Es razonable por lo que he explicado en la anterior pregunta. El usuario es el que pide el servicio en el servidor siempre, porque si no el servidor no sabe a quién dirigirse. Por lo tanto, en el primer paquete FTP, es en el de la bienvenida del Servidor. Si hablasemos de otros paquetes anteriores, sin conocer este, podría existir mayor duda y sería más intuitivo, pero no es el caso.

En algunos casos Wireshark no muestra las direcciones MAC completas sino que sustituye la primera mitad por una serie de caracteres. Investigue en Internet qué significan estos caracteres y cómo puede establecer Wireshark la correspondencia con cada dirección MAC.

Esa mitad de caracteres en concreto son los 6 primeros caracteres y lo que representa es el fabricante de la tarjeta de red. El ejemplo que encontramos en el primer paquete FTP es el siguiente:

"AsusNetw_04:c4:3a" es realmente "00:04:0f:04:c4:3a" y "AsusNetw_" equivale a "00:04:0f" y todos los dispositivos que tengan su tarjeta de red fabricada por AsusNetw empezarán por 00:04:0f.

Pregunta 4 (2 puntos)

Localice los paquetes en los que el cliente envía al servidor tanto el identificador de usuario como la contraseña de acceso. A nivel de FTP ¿Que mensajes envía el cliente y que respuestas obtiene del servidor en cada caso? Indique aquí el mensaje intercambiado en la comunicación (incluyendo el código correspondiente), no el texto que muestra Wireshark interpretando esos mensajes.

	Mensaje del cliente	Respuesta del servidor
Mensaje con el ID	Información del usuario	Requiere contraseña
Mensaje con la contraseña	Contraseña de Acceso	Inicio de sesión completado

¿Puede averiguar en estos mensajes el usuario y la contraseña de acceso utilizados? En caso afirmativo ¿cuales son?

◉ Si ID de usuario estudiante

○ No Contraseña Bonnie&Clyde

Pregunta 5 (3 puntos)

Localice en el listado los paquetes en los que se intercambian los mensajes **AUTH TLS** y **AUTH SSL**. Estos mensajes forman parte del diálogo para establecer la seguridad de la comunicación. No

vamos a entrar ahora en el detalle de los mismos pero, sin hacerlo, se puede responder a las siguientes preguntas. ¿Quien envía estos mensajes?

¿Quien envía estos mensajes? ◉ Cliente ○ Servidor ○ En blanco

¿Qué respuesta se obtiene de la otra parte de la comunicación? (con el código correspondiente)

502 Explicit TLS authentication not allowed

¿Cómo interpreta estas respuestas? ¿qué cree que significan?

La autentificación explícita TLS no está permitida. AUTH TLS y AUTH SSL son comandos dedicados a una comunicación segura. TLS es una versión mejorada de SSL, pero ambas tienen la misma función, protocolos de seguridad de la capa de transporte mediante un cifrado de mensajes para ocultarlos.

En el intercambio de información FTP se envía el contenido del directorio remoto que contiene tres ficheros. ¿Puede identificar los nombres de estos ficheros? Añada también el tamaño de cada uno.

Indique los nombres y tamaño (en bytes) de los ficheros que contiene el directorio remoto

Diapositiva_01.jpg 158112 bytes
Diapositiva_02.jpg 88189 bytes
Diapositiva_03.jpg 137608 bytes

¿Qué cree que sucedería si transfiriésemos cualquiera de esos ficheros a través de la red?

Que cualquier persona tendría acceso a esos archivos que transferiríamos. No habría ningún problema, lo que pasa es que otros usuarios podrían hacer un uso de esos archivos que no sean de tu gusto. Mediante protocolos y por tanto cifrados y seguridad evitamos este problema.

Pregunta 6 (1 punto)

Cargue ahora el fichero **Captura_FTP_cifrado.pcapng** correspondiente a la transferencia FTP cifrada. Revise las preguntas respondidas con respecto a la comunicación sin cifrar y describa a continuación las diferencias observadas. Responda de manera razonada a las siguientes preguntas: ¿Puede descubrir ahora el usuario y la contraseña? ¿Qué respuestas se obtienen ahora en el diálogo de seguridad? ¿Cómo interpreta ahora estas respuestas? ¿Puede descubrir ahora el contenido del directorio raíz remoto?

No se puede descubrir el usuario y la contraseña de manera aparente. La respuesta en el diálogo de seguridad es "234 Using authentication type TLS" que significa usando autentificación de tipo TLS. Los mensajes vienen encriptados y no se pueden encontrar los otros tres archivos de antes.

Una vez gestionada la autenticación ¿qué sucede con los posibles futuros paquetes del protocolo FTP? ¿Qué protocolo utilizan ahora el cliente y el servidor para intercambiar sus mensajes?

Utilizan un protocolo diferente evidentemente, que es el TLS v1.2 y la información que proporciona wireshark es que es " Application Data", interpreta qué tipo de información es pero no la información en concreto.

5 Parte 2: Protocolo ARP

En esta segunda parte vamos a realizar una breve práctica utilizando Wireshark para ver en un entorno real como trabaja el protocolo ARP. A diferencia de la primera parte, en esta segunda deberá realizar su propia captura. Posteriormente deberá enviar el archivo con esta captura junto con este documento convenientemente rellenado.

Con el objeto de reducir el número de peticiones ARP, todas las máquinas mantienen una tabla local con las últimas correspondencias utilizadas. Para ver esta tabla abra una ventana de comandos en Windows o una ventana de terminal en Linux y ejecute el comando **arp –a** . Para eliminar el contenido de la tabla ARP y forzar nuevas peticiones debe ejecutarse el comando **arp –d** (borra todas las entradas) **o arp –d <IP>** (borra la entrada correspondiente a la IP especificada). En Linux, para limpiar toda la tabla, debe utilizarse el comando **sudo ip neigh flush all**. En ambos sistemas se requieren privilegios de administrador para borrar la tabla ARP.

Si su máquina dispone de varias interfaces de red le aparecerán tantas tablas como interfaces. Para visualizar únicamente la tabla de una de las interfaces utilice el parámetro **-n** especificando la dirección IP vinculada con la interfaz: **arp –a –n <dir_IP>**.

Antes de comenzar conviene que recopile algunos datos de su red local y de dos servidores remotos (fuera de su red) independientes, por ejemplo **www.iana.org** y **www.wireshark.org**. Para obtener las direcciones IP de estos servidores elegidos puede utilizar el comando **ping**[2]. Una vez obtenida la IP de cada servidor remoto, utilice esta IP en lugar de su nombre.

	Dirección IP	Dirección MAC
Su máquina de trabajo	192.168.1.54	c6:39:89:ed:88:76
Dirección del rúter en la red local	192.168.1.1	2c-96-82-0d-8c-78

Anote aquí las direcciones IP de los servidores remotos elegidos y de un equipo local (impresora, SmartTV, tablet, teléfono móvil, etc)[3].

Dirección IP del servidor 1	192.0.43.8	
Dirección IP del servidor 2	104.26.11.240	
IP del otro equipo local	192.168.1.56	Ej. el teléfono móvil

Es importante utilizar las direcciones IP en lugar de los nombres de los servidores porque podrían formar parte de una CDN (Content Delivery Network) que puede utilizar IP’s distintas entre distintas peticiones a un mismo nombre de servidor y eso dificultaría el trabajo. Por este motivo, una vez identificada una IP válida para cada uno de los servidores elegidos, utilizaremos exclusivamente esas direcciones IP. También es importante que el equipo local esté conectado a la misma red local. Es indiferente si está conectado de manera cableada o inalámbrica.

El comando **ping** lo utilizaremos también para lanzar peticiones a cada uno de los servidores. Si hacemos esto después de limpiar la tabla ARP forzaremos una nueva petición ARP, lo que nos permitirá realizar la captura de la misma.

2 El comando **ping** permite enviar paquetes ICMP
3 Algunos dispositivos pueden estar configurados para no responder mensajes ICMP. Pruebe con otro.

Pregunta 7 (4 puntos)

Como ya habrá podido estudiar en el texto base, el protocolo ARP se encarga de proporcionarnos la dirección MAC correspondiente a una dirección IP. ¿Por qué necesita una máquina conocer la dirección MAC del destinatario si ya conoce su dirección IP?

Aunque una máquina conozca la dirección IP de un destinatario, necesita la dirección MAC para la entrega efectiva de paquetes en la red local, ya que las direcciones IP se utilizan a nivel de red y las direcciones MAC a nivel de enlace. El protocolo ARP facilita esta asociación dinámica entre direcciones IP y direcciones MAC, lo que se llama mapear direcciones IP.

Ahora realizaremos una captura sobre la que se centrarán las sucesivas preguntas. Para asegurar que se crean peticiones ARP, previamente deberemos hacer un borrado de la tabla ARP local. Realice los siguientes pasos en secuencia:

1. Inicie una captura de Wireshark
2. Borre la tabla ARP de su máquina
3. Ejecute un **ping –n 1 <IP>** a cada uno de los equipos elegidos previamente[4]
4. Espere a que lleguen las respuestas a las peticiones **ping** y pare la captura
5. Guarde la captura con el nombre **Nombre_Apellido1_Apellido2.pcapng**

Se recomienda realizar estos pasos en una secuencia rápida para reducir el número de paquetes capturados y simplificar su análisis. Para automatizar y simplificar esto se pueden poner los distintos comandos en un pequeño fichero BAT de manera que solo sea necesario ejecutar un comando. A modo de ejemplo el contenido de este archivo, en el caso de Windows, sería:

```
arp –d
ping –n 1 <IP servidor 1>
ping –n 1 <IP servidor 2>
ping –n 1 <IP equipo local>
arp –a
```

Consulte ahora la tabla ARP y busque las direcciones IP de los tres equipos elegidos, ¿Qué MAC tiene asociada cada uno y por qué? Explique su respuesta:

En la tabla ARP no aparecen ni las IPs ni las MAC de los dos primeros equipos elegidos (www.wikipedia.org y www.iana.org) porque son conexiones o comunicaciones que salen de nuestra red local y no están dentro de la capa de enlace, es decir del protocolo ARP. De hecho cuando realizamos el comando arp -a dentro del CMD no aparece ninguna de estas IPs. La IP y la MAC que aparece es la del equipo local (192.168.1.56) que tiene como MAC (b4:b2:91:01:1f:a6) que también aparece en wireshark en el protocolo ARP

4 El comando **ping** lanza una secuencia de peticiones. El parámetro **–n 1** indica que se realice únicamente una petición que es suficiente para nuestros propósitos. El algunos sistemas Linux, debe usarse el parámetro **–c** en lugar de **–n**

El comando **ping** utiliza el protocolo ICMP. Filtre por este protocolo para visualizar las peticiones y respuestas de este comando. En estos paquetes identifique las direcciones MAC a las que se envían (y de las que se reciben) los paquetes de cada uno de los servidores web:

MAC a la que se envían las peticiones (y se reciben las respuestas) en el caso del servidor 1	2c:96:82:0d:8c:78
MAC a la que se envían las peticiones (y se reciben las respuestas) en el caso del servidor 2	2c:96:82:0d:8c:78
MAC a la que se envían las peticiones (y se reciben las respuestas) en el equipo local	b4:b2:91:01:1f:a6

¿Se parecen en algo estas direcciones MAC? ¿A quien corresponden cada una de estas direcciones MAC? Explique con detalle por qué cree que sucede esto:

Las dos primeras MAC son idénticas que la MAC del Router porque es el último contacto tanto de emisor como de receptor del protocolo ICMP, del que podemos saber su MAC, ya que está dentro de nuestra red local. La MAC del dispositivo elegido coincide con su propia MAC consultada en la información de ese dispositivo y nos aparece esa MAC porque está dentro de nuestra red local y es el último dispositivo en la comunicación tanto de emisor como de receptor.

Pregunta 8 (3 puntos)

Localice, en la captura realizada, una pareja de petición–respuesta de ARP. Como la captura puede contener más de una pareja de este tipo, indique los números de paquete de la pareja elegida como referencia para preguntas posteriores.

N.º paquete petición	57
N.º paquete respuesta	58

A continuación calcule el tiempo transcurrido entre la petición y la respuesta:

Tiempo transcurrido entre la petición y la respuesta: (Especifique las unidades temporales correctas)	0.001834000 segundos

Puede haber múltiples parejas de este tipo en una misma captura. Explique cómo puede realizar el emparejamiento correcto, atendiendo exclusivamente al contenido de los paquetes sin tener en cuenta la ayuda que proporciona Wireshark:

Wireshark se encarga de descifrar de una manera más accesible o entendible el código de los paquetes. Por lo tanto, con este código,sin la herramienta Wireshark podemos ir descifrando la información por nosotros mismos aunque es un poco tedioso. Por lo tanto, en el código alberga quién envia el mensaje en forma de IP y MAC y a quién va a llegar el mensaje (IP y MAC) que se tiene que encargar de responder al mesaje ya que es una request.

Si no contásemos con la asistencia de Wireshark para interpretar el contenido de los paquetes, atendiendo al contenido binario del paquete, ¿cómo podríamos identificar los paquetes que corresponden al protocolo ARP?

Leyendo el código binario del paquete encontramos que después del apartado de las MAC encontramos un "0806", que corresponde al uso del protocolo ARP.

Teniendo en cuenta que el primer byte del paquete está en la posición 0. Indique la posición, dentro de todo el paquete capturado, de los bytes que nos proporcionan esa información:

Posiciones de los bytes que nos indican que se trata de un paquete correspondiente al protocolo ARP	12-13

Pregunta 9 (4 puntos)

Rellene los campos del protocolo ARP correspondientes a la pareja Petición–Respuesta elegida previamente.

	Petición	Respuesta
Tipo dirección de hardware (valor y significado)	1 (Ethernet)	1 (Ethernet)
Tipo de red/protocolo (valor y significado)	x800 (IPv4)	x800 (IPv4)
Longitud dirección física	6 bytes	6 bytes
Longitud dirección lógica	4 bytes	4 bytes
Tipo operación	Request	Reply
MAC remitente	c6:39:89:ed:88:76	2c:96:82:0d:8c:78
IP remitente	192.168.1.54	192.168.1.1
MAC objetivo	ff:ff:ff:ff:ff:ff	c6:39:89:ed:88:76
IP destinatario	192.168.1.1	192.168.1.54

Pregunta 10 (3 puntos)

Localice el paquete de petición de la pareja petición–respuesta ARP especificada previamente y responda a las siguientes preguntas:

¿quien es el emisor del paquete y cual es su MAC? (indique si se trata de su ordenador de trabajo, el rúter, un servidor externo, otro equipo de la misma red, etc.)

¿Qué máquina envía la petición ARP?	192.168.1.54
¿Cual es su dirección MAC?	c6:39:89:ed:88:76

¿a qué dirección MAC va destinada la petición y a quien corresponde esa dirección?

¿MAC de destino de la trama ARP?	ff:ff:ff:ff:ff:ff
¿a quien corresponde?	Broadcast, para buscar qué MAC corresponde a la IP de esa red

¿De qué dirección IP se pretende obtener la dirección MAC y en qué posición del paquete viaja esta información?

Dirección IP	192.168.1.1
Bytes en los que va esta información	38-41

6 Comentarios, conclusiones

Indique aquí las conclusiones y comentarios que considere oportuno. Este apartado es simplemente una realimentación y no tiene carácter evaluador.