ACTIVIDAD 2: ANÁLISIS DE TRAMAS DE UN ACCESO HTTP - WIRESHARK

Seguridad y Alta Disponibilidad

19 DE SEPTIEMBRE DE 2024

JUAN PABLO ORTOLÁ VILLANUEVA
U-TAD Curso 24-25

Índice

1. Introducción

La seguridad de la información ha llegado a niveles muy elevados y cada vez es más necesaria para cualquier tipo de sistema que tenga conexión a Internet, más aún si diariamente se trabaja en un entorno donde se maneja información sensible. Actualmente, las conexiones son más seguras a través del protocolo HTTPS que permite un cifrado de datos entre usuario y servidor.

Aún así, hay sitios web que aún utilizan el protocolo HTTP que, a diferencia del anterior, no permite un cifrado de la información y toda la información se ve de manera clara y sin ningún tipo de seguridad o privacidad.

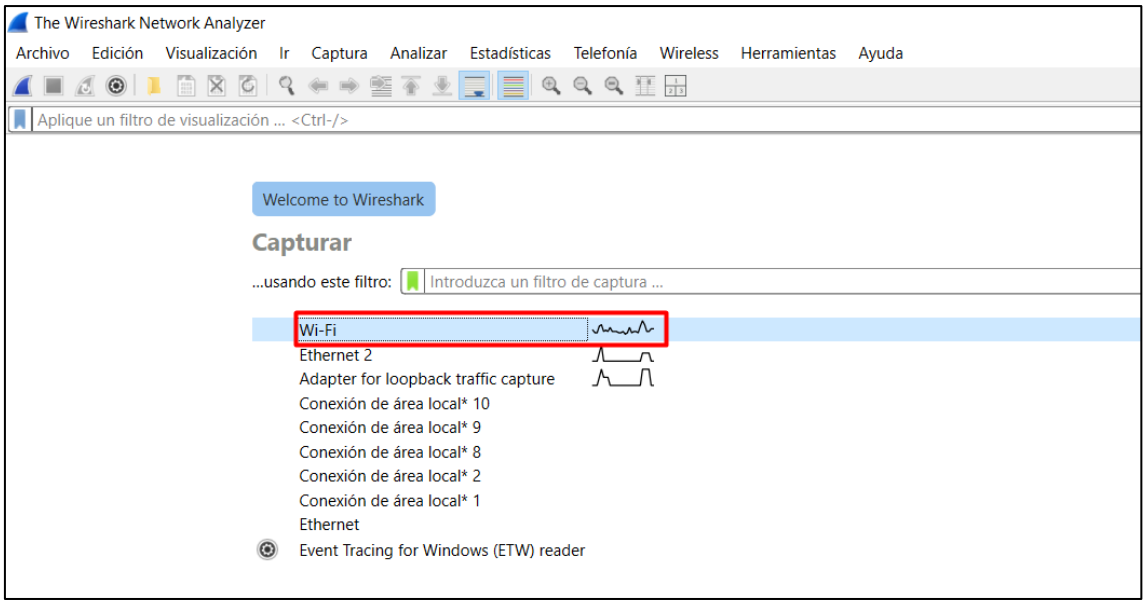
Este proyecto utilizará Wireshark, una herramienta que realiza la captura y el análisis de paquetes de red, para analizar de manera detallada las vulnerabilidades comunes que presenta el protocolo HTTP. Esto lo podremos ver ya que accederemos a la página http://celfi.gob.ar y nos identificaremos mediante un *login* para poder ver dichas vulnerabilidades.

2. Wireshark

Wireshark es una herramienta de análisis que permite capturar y examinar el tráfico de datos en tiempo real. Principalmente, esta aplicación nos permite diagnosticar problemas que puedan ocurrir en la red, realizar análisis de protocolos y detectar vulnerabilidades de seguridad.

Durante la instalación, nos pedirán que aceptemos algunos parámetros o si queremos instalar algunos paquetes extras para poder realizar algunas tareas extras. La más importante, es habilitar el *npcap* ya que es una librería y controlador de red necesario para capturar los paquetes de red en Windows. Tiene una serie de funciones que lo hacen esencial para Wireshark, a parte de la captura de paquetes, es compatible con varios protocolos distintos, mejora el rendimiento y es capaz de capturar todo el tráfico que pasa por la red y no solo el que pasa por la máquina que lo ejecuta.

Como vemos en la imagen de abajo (img. 1), antes de empezar a analizar el tráfico de red, tendremos que elegir una interfaz de red. El hecho de que haya varias opciones para elegir es debido a que están representados los diferentes dispositivos de red por el cual se puede realizar un intercambio de datos. Nos quedaremos con la opción Wi-Fi y capturaremos los paquetes que viajan por el aire desde dispositivos que estén conectados a la misma red.

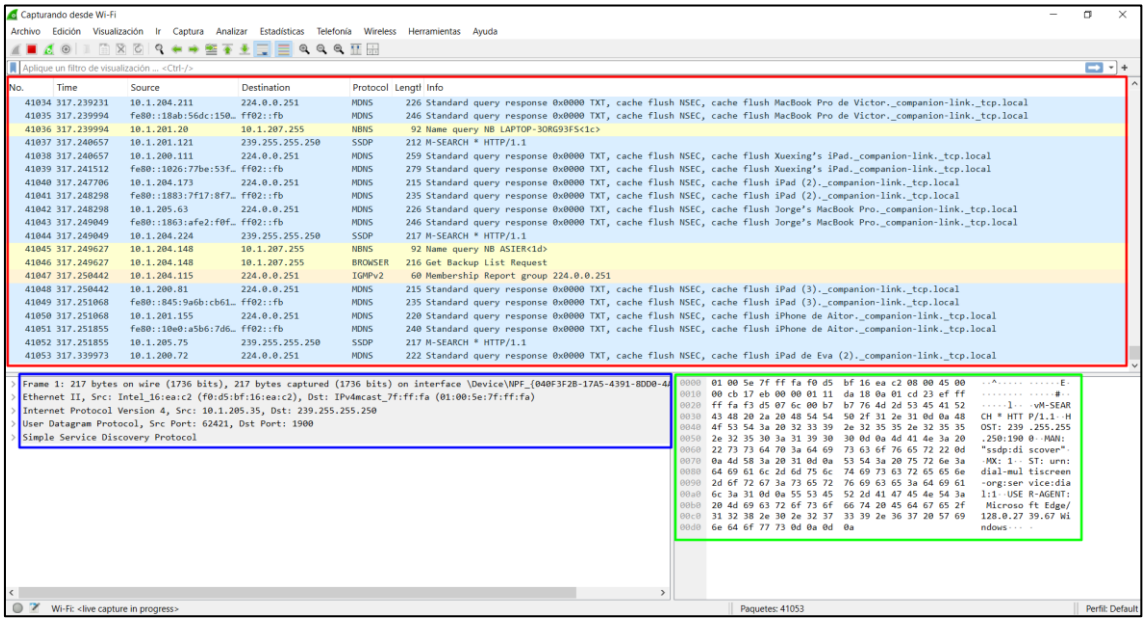


Img. 1

3. Análisis de tramas general

Antes de empezar con el análisis de tramas propuesto para este proyecto, haremos un análisis general para poder entender que nos aporta la interfaz del Wireshark. En la parte superior de la (img. 2) del lado izquierdo podemos ver el símbolo de una aleta y otro de un cuadrado rojo, uno permite comenzar a analizar las tramas y el otro parar el proceso, respectivamente.

- El cuadrado rojo nos indica la lista de paquetes en un modo de lista y en el orden en que han sido capturados. Cada fila corresponde a un paquete y se pueden ver detalles como el número, tiempo, origen, destino, protocolo e incluso una breve descripción.
- El cuadrado azul muestra los detalles del paquete que se abre al hacer click en alguno de ellos y muestra de manera detallada un desglose de las capas (Ethernet, IP, TCP/IP, etc). Además, hay información como direcciones IP, puertos y contenido específico del protocolo.
- Por último, el cuadrado verde nos enseña datos del paquete, pero en modo bruto, el contenido viene en un formato hexadecimal y ASCII y permite ver en detalle a nivel de bits de tráfico de red.



Img. 2

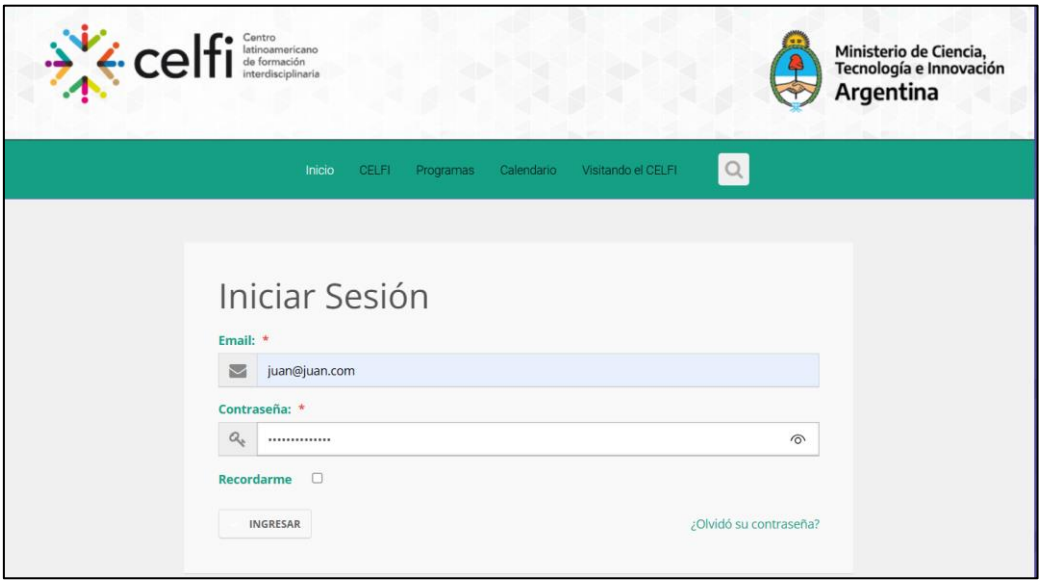
4. Análisis de tramas (HTTP)

Como hemos explicado anteriormente, el objetivo de este proyecto es analizar la seguridad de una conexión HTTP (Hypertext Transfer Protocol) no segura. A diferencia de HTTPS (Hypertext Transfer Protocol ***Secure***), este protocolo nos permite ver sin ningún cifrado cualquier tipo de información sensible que se haya escrito en la web o esté ocultada en la misma.

El protocolo HTTPS cuenta con una tecnología denominada SSL (Secure Sockets Layer), también conocidos como certificados digitales, y se utilizan para establecer una conexión cifrada entre un navegador y un servidor web.

Ahora que sabemos cómo funciona la aplicación, vamos a poder buscar estas vulnerabilidades que tiene el protocolo HTTP. Lo conseguiremos metiéndonos en la página http://celfi.gob.ar (img. 3), mientras nuestro Wireshark está capturando tráfico de red, e iniciando sesión como en el ejemplo puesto en la (img. 4).

Img. 3

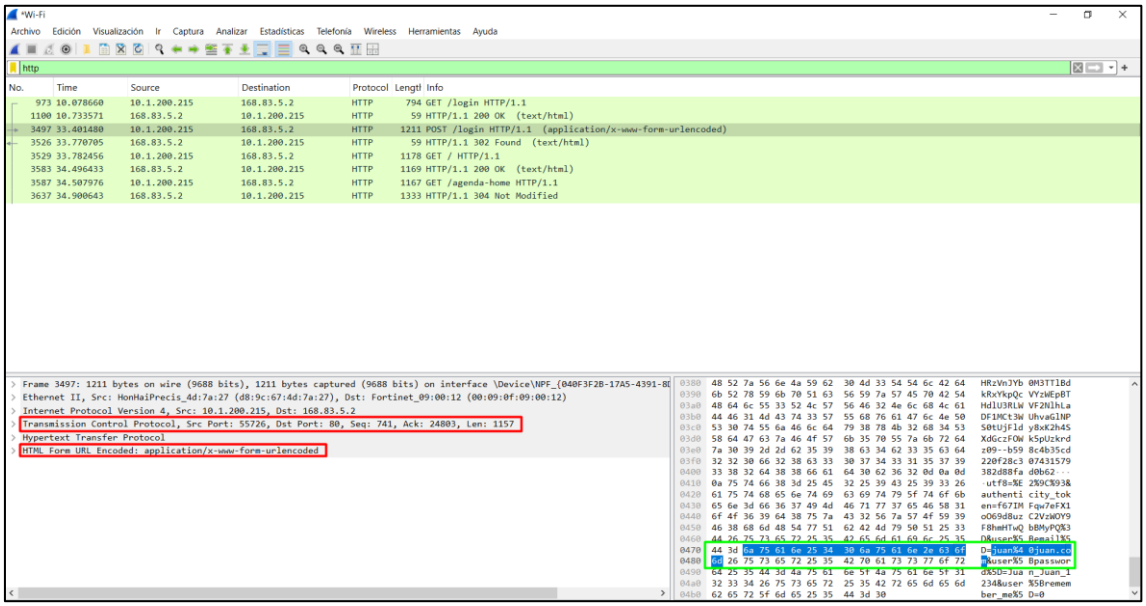


Img. 4

Cuando terminamos de iniciar sesión y ponemos en la barra de búsqueda del Wireshark “http” nos debería quedar una imagen como la que vemos a continuación (img. 5). Hay uno de los paquetes donde podremos ver claramente toda la información que estamos buscando, está marcado de color gris y se puede identificar por la frase:

POST/login HTTP/1.1 (application/x-www-form-urlencoded)

En la parte de debajo del lado izquierdo podremos ver los detalles del paquete y las diferentes capas del modelo OSI, representadas con una flecha y las que más nos interesan son las marcadas en rojo. Explicaremos brevemente las capas en este ejemplo.

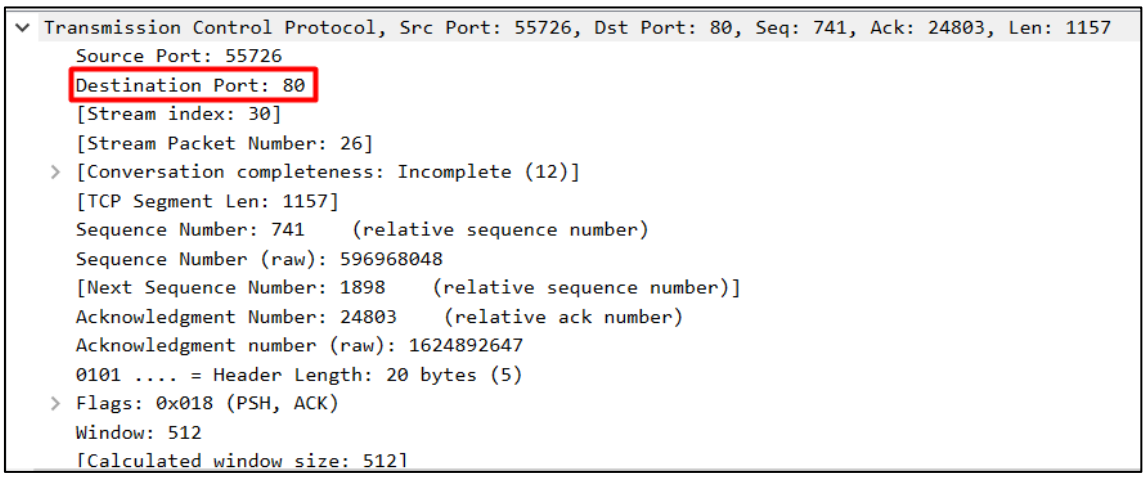


Img. 5

- **Capa Física (1):** esta capa nos indica el tamaño del paquete en bits y bytes, y por donde fue capturado, es decir, el medio por el que se transmitió la información (cable, fibra, etc).
- **Capa de Enlace de Datos (2):** dicha capa identifica el protocolo Ethernet II, que se encarga de enmarcar los datos para su transmisión en una red local.
- **Capa de Red (3):** se encarga de indicar que se está utilizando el protocolo IP versión 4 (IPv4), que procura enrutar los paquetes a través de múltiples redes. En nuestro caso la dirección IP de origen es 10.1.200.215 y la de destino 168.83.5.2.
- **Capa de Transporte (4):** este utiliza el protocolo TCP (Transmission Control Protocol), así garantiza que la entrega de datos sea fiable y ordenada. Nuestro

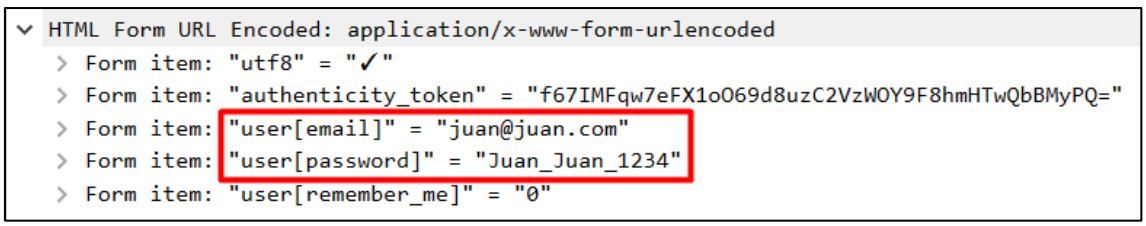
puerto de origen es el 55726 y el de destino el puerto 80, que es el puerto estándar para el protocolo HTTP y hablaremos más de él en el siguiente punto.

- **Capas Superiores (5, 6, 7):** las últimas dos líneas nos indican que el protocolo HTTP está siendo utilizado y se encarga de la comunicación entre el cliente y el servidor.



Img. 6

El puerto 80 (img. 6) es el puerto predeterminado para HTTP, como hemos dicho anteriormente, es el más básico para la transferencia de páginas web e información por la web. Usar este puerto te hace vulnerable a ataques como *sniffing* o *man-in-the-middle-attacks* donde los atacantes podrían interceptar o modificar la comunicación. Es un protocolo que no cifra sus datos y en su momento era el principal formato de tránsito de información por la web. Actualmente con la llegada del HTTPS (puerto 443), que ofrece mucha más seguridad y cifrado, el uso de HTTP es casi nulo. Estos puertos están designados por IANA (Internet Assigned Numbers Authority), una organización que gestiona los espacios de direcciones IP y los números puertos.



Img. 7

Esta última imagen (img. 7), es justo lo que necesitamos para el objetivo del proyecto. Como podemos ver, en la capa OSI (5, 6, 7), explicadas en el punto anterior, la información que utilizamos al iniciar sesión en la página web no está cifrada y es claramente legible. Aquí vemos la poca seguridad que aporta el protocolo HTTP y la razón por la que cada vez se utiliza menos.

5. Conclusiones

Este proyecto nos ha dejado identificar de manera clara los riesgos que lleva utilizar el protocolo HTTP dando evidencias de que las credenciales y otros datos sensibles pueden ser interceptados con facilidad al transmitirse en texto claro.

El uso de Wireshark resulta ser muy útil para poder analizar y anotar todas las vulnerabilidades que presenta el HTTP y comprender la necesidad de implementar el protocolo HTTPS, ofreciendo mayor seguridad y cifrado gracias a su tecnología SSL.

Como última anotación, cabe destacar la importancia que tiene en la formación de ciberseguridad usar estas herramientas de análisis de red para auditar y mejorar la seguridad de las comunicaciones en cualquier entorno.

6. Bibliografía

- OpenWebinars. (n.d.). *Wireshark: Qué es y ejemplos de uso*. OpenWebinars. https://openwebinars.net/blog/wireshark-que-es-y-ejemplos-de-uso/
- DigiCert. (n.d.). *¿Qué es SSL, TLS y HTTPS?* DigiCert. https://www.digicert.com/es/what-is-ssl-tls-and-https