## Documentación del Proyecto: Captura de Tráfico HTTP no Cifrado con Wireshark

## 1. Resumen Ejecutivo

## Este documento detalla el procedimiento para configurar un entorno de red virtual, establecer un servidor web local y capturar y analizar el tráfico de una solicitud HTTP no cifrada. El objetivo principal fue demostrar la vulnerabilidad de la información transmitida a través de un protocolo sin cifrado, como el HTTP, utilizando el analizador de protocolos Wireshark.

## 2. Objetivos del Proyecto

## Configurar una red aislada: Establecer una conexión de red entre dos máquinas virtuales (Ubuntu y Kali Linux) utilizando una Red Interna de VirtualBox.

## Crear un servidor web no seguro: Poner en marcha un servidor HTTP local que sirva contenido en texto plano.

## Capturar el tráfico de red: Utilizar Wireshark para interceptar todos los paquetes de datos que circulan entre el servidor y el cliente.

## Analizar la información en texto plano: Demostrar que los datos de una solicitud y respuesta HTTP son visibles y legibles para cualquier persona que capture el tráfico.

## 3. Herramientas y Entorno

## Software de Virtualización: VirtualBox

## Máquina Virtual 1 (Servidor): Ubuntu

## Máquina Virtual 2 (Cliente / Analizador): Kali Linux

## Servidor Web: Python 3 (http.server)

## Analizador de Protocolos: Wireshark

## 4. Procedimiento Detallado

## Paso 1: Configuración del Entorno Virtual

## Ambas máquinas virtuales se configuraron en VirtualBox con el adaptador de red en modo "Red Interna". Esto aseguró que pudieran comunicarse entre sí en una red aislada, sin acceso al exterior.

## Paso 2: Levantamiento del Servidor HTTP (Ubuntu)

## Se creó un archivo HTML de prueba (test.html) en la carpeta principal de Ubuntu.

## Bash

## echo "<h1>¡El servidor de prueba funciona!</h1>" > test.html

## Se levantó un servidor web en el puerto 8000 con el módulo de Python.

## Bash

## python3 -m http.server 8000

## Se verificó la dirección IP de la máquina para la red interna con el comando ifconfig, identificándose como 10.0.3.5 en la interfaz enp0s8.

## Paso 3: Captura de Paquetes (Kali Linux)

## Se identificó la interfaz de red de Kali conectada a la red interna, que resultó ser eth1 con la dirección IP 10.0.3.4.

## Se inició Wireshark con permisos de superusuario para asegurar que pudiera capturar todos los paquetes.

## Bash

## sudo wireshark

## Dentro de Wireshark, se seleccionó la interfaz eth1 para comenzar la captura.

## Paso 4: Generación y Análisis del Tráfico

## En el navegador de Kali (Firefox), se accedió a la página servida por Ubuntu.

## http://10.0.3.5:8000/test.html

## Tras la carga de la página, la captura en Wireshark se detuvo.

## Se aplicó un filtro para aislar el tráfico relevante.

## tcp.port == 8000 and http

## Se seleccionó el paquete GET /test.html y se utilizó la función Follow > HTTP Stream.

## 5. Conclusiones y Demostración de Vulnerabilidad

## El ejercicio demostró de manera contundente la naturaleza no cifrada del protocolo HTTP. En la ventana de "HTTP Stream" de Wireshark, se pudo observar el contenido de la solicitud y la respuesta, incluyendo el mensaje <h1>¡El servidor de prueba funciona!</h1>, expuesto en texto plano.

## Esto evidencia que cualquier información sensible (como nombres de usuario, contraseñas o datos personales) enviada a través de un sitio web con HTTP sería completamente visible para un atacante que esté capturando el tráfico en la misma red. La única forma de proteger esta información es migrando a protocolos de comunicación seguros como HTTPS, que encriptan el tráfico de extremo a extremo.