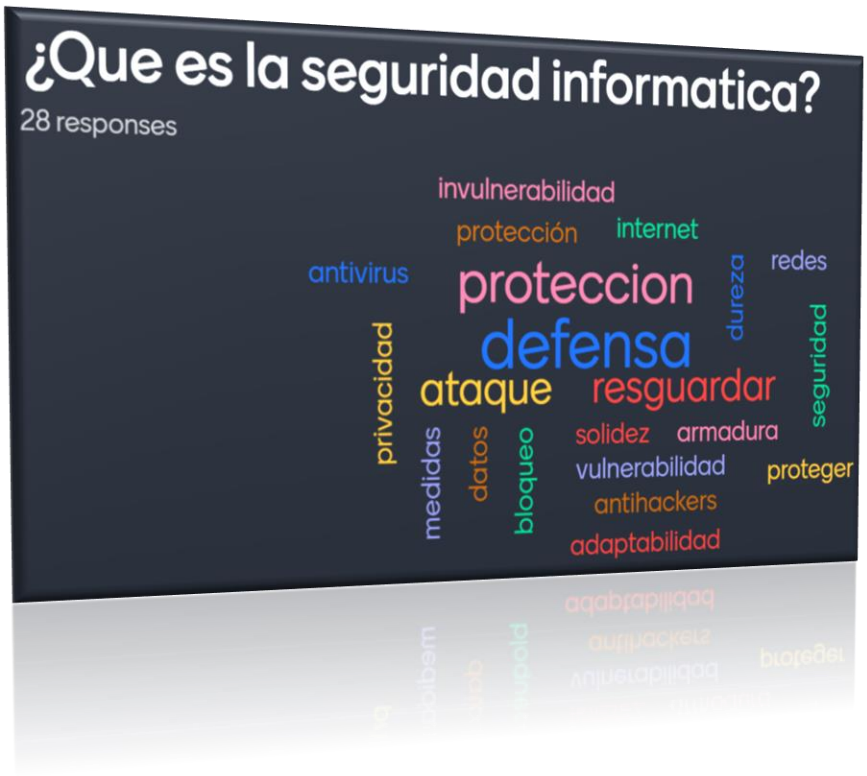
Universidad Paraguayo Alemana

UNIVERSIDAD PARAGUAYO ALEMANA
HEIDELBERG - ASUNCIÓN



Seguridad TICs

Prof.: Chrystian Ruiz Diaz

Contenido

Nota de Uso Académico

Este documento ha sido preparado exclusivamente para el uso académico del docente. Este documento no puede ser distribuido a ninguna otra parte ni utilizado para ningún otro propósito, diferente al establecido como alcance en el proyecto académico de la **UNIVERSIDAD PARAGUAYO ALEMANA**. El uso indebido del material fuera del ámbito académico no representa ninguna responsabilidad del docente.

Sniffer - Wireshark

Introducción a Wireshark

Wireshark es una herramienta de análisis de protocolos de red ampliamente utilizada para capturar y examinar los datos que se transmiten a través de una red informática en tiempo real. Esta herramienta es invaluable para administradores de red, ingenieros de seguridad, desarrolladores y educadores que necesitan entender el funcionamiento interno de las redes y diagnosticar problemas.

¿Qué es Wireshark?

Wireshark es un analizador de paquetes de red que permite a los usuarios ver todo el tráfico que pasa por su red. Es de código abierto y está disponible de forma gratuita. Wireshark puede capturar datos en tiempo real y guardarlos para análisis posterior. Permite a los usuarios explorar en detalle cada paquete de datos que se mueve a través de la red.

Historia y Desarrollo

Wireshark fue creado por Gerald Combs en 1998 bajo el nombre de Ethereal. En 2006, debido a problemas de marca registrada, el proyecto fue renombrado a Wireshark. Desde entonces, Wireshark ha crecido en popularidad y es considerado el estándar de facto para el análisis de tráfico de red.

¿Para qué se Usa Wireshark?

Wireshark se utiliza para una variedad de propósitos, incluyendo:

1. **Resolución de Problemas de Red**: Identificar cuellos de botella, problemas de conectividad y errores de configuración.
2. **Análisis de Seguridad**: Detectar y analizar actividades sospechosas y ataques en la red.
3. **Desarrollo y Pruebas de Software**: Analizar el tráfico generado por aplicaciones para asegurar que se comporten según lo esperado.
4. **Educación**: Enseñar sobre protocolos de red y análisis de tráfico a estudiantes y profesionales.

Características de Wireshark

Wireshark ofrece una amplia gama de características que lo hacen extremadamente poderoso:

1. **Captura de Datos en Tiempo Real**: Puede capturar el tráfico de red en tiempo real desde varias interfaces.
2. **Soporte de Protocolos Extensivo**: Wireshark soporta más de 2000 protocolos diferentes.

3. **Filtros de Captura y Visualización**: Permite aplicar filtros para capturar o visualizar solo el tráfico de interés.
4. **Análisis Detallado de Protocolos**: Desglosa y presenta cada paquete en un formato legible, con detalles sobre cada campo.
5. **Reconstrucción de Sesiones TCP**: Puede ensamblar y mostrar sesiones completas de comunicaciones TCP.
6. **Desempaquetado de Datos en Varias Capas**: Muestra datos desde la capa física hasta la aplicación.
7. **Soporte para Múltiples Plataformas**: Disponible para Windows, macOS y Linux.
8. **Exportación de Datos**: Permite exportar datos en diferentes formatos para análisis adicionales.

Explicación de las Funciones con Ejemplos

1. Captura de Paquetes

Para capturar paquetes, selecciona la interfaz de red apropiada y haz clic en el botón de inicio de captura. Wireshark comenzará a registrar todo el tráfico que pasa por esa interfaz.

Ejemplo:

1. Abre Wireshark.
2. Selecciona la interfaz de red (e.g., `Wi-Fi`, `Ethernet`).
3. Haz clic en el botón azul con un tiburón para iniciar la captura.
4. Observa cómo se llenan los datos en tiempo real.

2. Aplicación de Filtros

Wireshark permite aplicar filtros para enfocar el análisis en tráfico específico.

Filtros de Captura: Se aplican antes de iniciar la captura y filtran el tráfico capturado.

- Capturar solo tráfico HTTP:

```
port 80
```

- Capturar solo tráfico desde o hacia una IP específica:

```
host 192.168.1.1
```

Filtros de Visualización: Se aplican después de que el tráfico ha sido capturado y filtran el tráfico visible en la interfaz.

- Ver solo tráfico HTTP:


```
http
```

- Ver tráfico desde una IP específica:

```
ip.addr == 192.168.1.1
```

- Ver paquetes TCP:

```
tcp
```

- Ver paquetes DNS:

```
dns
```

- Ver tráfico en un puerto específico:

```
tcp.port == 443
```

Operadores Comunes en Filtros de Visualización

- **AND (`&&` o `and`)**: Este operador se utiliza para combinar varios criterios y solo mostrar paquetes que cumplan con todas las condiciones.
  - Ejemplo: Mostrar solo paquetes TCP desde una IP específica.

```
ip.addr == 192.168.1.1 and tcp
```

- **OR (`||` o `or`)**: Este operador se utiliza para combinar varios criterios y mostrar paquetes que cumplan con al menos una de las condiciones.
  - Ejemplo: Mostrar paquetes que sean TCP o UDP.

```
tcp or udp
```

- **IGUAL (`==`)**: Este operador se utiliza para comparar campos y mostrar paquetes que coincidan exactamente con el valor especificado.
  - Ejemplo: Mostrar paquetes de un puerto específico.

```
tcp.port == 80
```


- **NO (`!` o `not`)**: Este operador se utiliza para excluir paquetes que coincidan con una condición específica.
    - Ejemplo: Mostrar todos los paquetes excepto los de una IP específica.

```
not ip.addr == 192.168.1.1
```

3. Análisis de Protocolos

Wireshark desglosa los paquetes en sus componentes de protocolo. Al hacer clic en un paquete capturado, puedes ver una representación detallada de cada capa del paquete.

Ejemplo:

- Selecciona un paquete HTTP.
- En el panel inferior, observa las capas: Ethernet II, IP, TCP y HTTP.
- Expande cada capa para ver detalles como direcciones MAC, IPs, puertos, y datos del protocolo HTTP.

4. Reconstrucción de Sesiones

Wireshark puede reconstruir sesiones completas de comunicaciones, lo que es útil para seguir una conversación TCP completa.

Ejemplo:

- Captura tráfico de una sesión web.
- Usa el menú `Seguimiento` > `Flujo TCP` para ver la conversación completa.

5. Búsqueda dentro de Paquetes

Wireshark permite realizar búsquedas dentro de los paquetes capturados para encontrar datos específicos.

Ejemplo:

- Captura tráfico de red.
- Usa `Ctrl + F` para abrir la ventana de búsqueda.
- Puedes buscar por datos de la capa específica, como `String`, `Hex Value`, o `Display Filter`.
- Buscar una cadena específica en los datos:

```
GET /index.html
```


6. Exportación de Datos

Wireshark permite exportar datos capturados para análisis adicional o para compartir con otros.

Ejemplo:

- Captura algunos paquetes.
- Ve a `Archivo` > `Exportar paquetes capturados` para guardar los datos en un archivo `.pcap` o `.txt`.

Ejercicios Prácticos

Ejercicio 1: Captura Básica

Objetivo: Aprender a capturar tráfico de red.

Pasos:

1. Abre Wireshark y selecciona la interfaz de red que deseas monitorear (e.g., `Wi-Fi`, `Ethernet`).
2. Haz clic en el botón azul con un tiburón para iniciar la captura.
3. Navega por Internet durante 5 minutos. Puedes visitar algunos sitios web, realizar búsquedas y observar cómo se capturan los paquetes.
4. Después de 5 minutos, haz clic en el botón rojo cuadrado para detener la captura.
5. Guarda el archivo de captura seleccionando `Archivo` > `Guardar como` y elige un formato como `.pcap`.

Ejercicio 2: Filtros de Captura

Objetivo: Aprender a aplicar filtros de captura para enfocar el análisis en tráfico específico.

Pasos:

1. Abre Wireshark y selecciona la interfaz de red.
2. Antes de iniciar la captura, aplica un filtro para capturar solo tráfico HTTP. En el campo de filtro de captura, escribe:

```
port 80
```

3. Haz clic en el botón azul con un tiburón para iniciar la captura.
4. Navega a algunos sitios web y observa que solo se capturan paquetes HTTP.
5. Detén la captura después de unos minutos y guarda el archivo.

Ejercicio 3: Análisis de Paquetes

Objetivo: Aprender a analizar los detalles de un paquete específico.

Pasos:

1. Abre Wireshark y selecciona la interfaz de red.
2. Captura tráfico de tu red doméstica durante unos minutos.
3. Detén la captura y usa el campo de filtro de visualización para encontrar paquetes DNS. Escribe:

```
dns
```

4. Selecciona un paquete DNS y en el panel inferior, expande las capas para analizar los campos del paquete DNS. Describe los campos como la consulta, la respuesta, el nombre del dominio, etc.

Ejercicio 4: Seguimiento de Sesión

Objetivo: Aprender a reconstruir y visualizar sesiones de comunicación completas.

Pasos:

1. Abre Wireshark y selecciona la interfaz de red.
2. Captura tráfico mientras descargas un archivo grande.
3. Detén la captura una vez que la descarga haya finalizado.
4. Usa la función de seguimiento de flujo TCP:
   - Selecciona un paquete de la descarga.
   - Haz clic derecho y selecciona `Seguir` > `Flujo TCP`.
   - Observa la conversación completa entre el cliente y el servidor.
5. Analiza los datos reconstruidos y describe el proceso de transferencia.

Ejercicio 5: Exportación y Compartición

Objetivo: Aprender a exportar datos capturados para análisis adicional.

Pasos:

1. Captura tráfico durante una videollamada breve.
2. Detén la captura y guarda el archivo.
3. Ve a `Archivo` > `Exportar paquetes capturados`.
4. Selecciona el formato `.pcap` y guarda el archivo.
5. Comparte el archivo con un colega para un análisis conjunto.

Ejercicio 6: Reconstrucción de un Archivo Descargado

Objetivo: Aprender a reconstruir un archivo descargado (e.g., un archivo ISO) a través de HTTP.

Pasos:

1. **Captura del Tráfico de Red**:
   - Abre Wireshark y selecciona la interfaz de red adecuada.
   - Configura el filtro de captura `port 80`.
   - Haz clic en el botón azul con un tiburón para comenzar la captura.
   - Descarga el archivo ISO desde una página web.
   - Detén la captura una vez que la descarga haya finalizado.
2. **Filtrado del Tráfico HTTP**:
   - Aplica el filtro `http` para mostrar solo el tráfico HTTP capturado.
3. **Identificación del Tráfico de Descarga**:
   - Busca los paquetes HTTP que contienen partes del archivo ISO, generalmente con el método `GET`.
   - Haz clic derecho en uno de los paquetes relacionados con la descarga y selecciona `Seguir` > `Flujo TCP`.
4. **Exportación de los Datos**:
   - Ve a `Archivo` > `Exportar objetos` > `HTTP`.
   - Encuentra el archivo ISO en la lista de objetos HTTP capturados y selecciónalo.
   - Haz clic en `Guardar` para exportar el archivo ISO.
5. **Verificación del Archivo**:
   - Utiliza herramientas como `md5sum` o `sha256sum` para verificar la integridad del archivo exportado.

Ejemplo Práctico:

1. **Captura del Tráfico**:
   - Abre Wireshark y selecciona la interfaz de red (e.g., `Wi-Fi`).
   - Aplica el filtro de captura `port 80`.
   - Inicia la captura y descarga el archivo ISO desde un sitio web.
   - Detén la captura una vez finalizada la descarga.
2. **Filtrado y Seguimiento del Tráfico**:
   - Aplica el filtro de visualización `http`.
   - Identifica los paquetes `GET` relacionados con el archivo ISO.
   - Haz clic derecho y selecciona `Seguir` > `Flujo TCP`.
3. **Exportación del Archivo**:
   - Ve a `Archivo` > `Exportar objetos` > `HTTP`.
   - Selecciona el archivo ISO y haz clic en `Guardar`.
4. **Verificación del Archivo**:
   - Utiliza `md5sum` o `sha256sum` para verificar la integridad del archivo exportado.

Recursos Adicionales

Para más información y recursos sobre Wireshark, puedes visitar:

https://www.wireshark.org/

Este material te proporciona una base sólida para entender y utilizar Wireshark, junto con ejercicios prácticos para afianzar tus conocimientos.

