

Seguridad y Auditoría Informática

Guía de prácticos TCP/IP

| Ing. Fernando Boiero | Año 2024 |
| --- | --- |

**Tabla de Contenidos**

# 

[**Guía de Trabajos Prácticos 3**](#_heading=h.am2u6fj7r08h)

[**Herramientas y Configuraciones Iniciales 3**](#_heading=h.dop7budk59g2)

[Software Requerido 3](#_heading=h.2ai9iufmmacn)

[Wireshark 3](#_heading=h.hpyjzpsvu70z)

[GNS3 3](#_heading=h.1npxqrvcemhq)

[Packet Tracer 3](#_heading=h.53enq0fxkb8o)

[Nmap 3](#_heading=h.ejak91mcczih)

[Metasploit Framework 3](#_heading=h.us421er6vu23)

[TCPdump 3](#_heading=h.jyherfcm35m3)

[Actividades Prácticas 4](#_heading=h.kkluesswb87v)

[Actividad 1: Captura y Análisis de Tráfico con Wireshark 4](#_heading=h.9gpy04zd6tt8)

[Objetivos 4](#_heading=h.ju3lpr8v0xtw)

[Pasos Detallados 4](#_heading=h.3h1makpxh6sj)

[Referencias Teóricas 5](#_heading=h.va951cv23ab8)

[Pasos Adicionales para la Exploración 5](#_heading=h.jjjghj3p4p8f)

[**Actividad 2: Diseño y Configuración de Redes con GNS3 6**](#_heading=h.9ujlavsd609o)

[Objetivos 6](#_heading=h.fbp366aqwsx2)

[Pasos Detallados 6](#_heading=h.25t5ngn2hmg1)

[Referencias Teóricas 7](#_heading=h.igfu97o2jd7y)

[Pasos Adicionales para la Exploración 7](#_heading=h.kp2h4y629ean)

[**Actividad 3: Simulación de Ataques y Defensas en Red 8**](#_heading=h.e7eluiuhsty8)

[Objetivos 8](#_heading=h.gzhhf1f40m8u)

[Pasos Detallados 8](#_heading=h.2h6gprd56he3)

[Referencias Teóricas 9](#_heading=h.m3ed2snvi5p0)

[Pasos Adicionales para la Exploración 9](#_heading=h.mrvqjuadkmfe)

[Trabajo Integrador: Auditoría de Seguridad de una Red 10](#_heading=h.kiw7han98csa)

[Objetivo 10](#_heading=h.jio2z684bkfq)

[Descripción del Proyecto 10](#_heading=h.hpifdv46gyux)

[Referencias Bibliográficas 10](#_heading=h.sszbvdo0vkux)

[**APÉNDICE 1- Generación de Servicios con Vulnerabilidades Específicas 11**](#_heading=h.y6yfylfkb3lq)

[Configuración de Servicios Vulnerables 11](#_heading=h.hatz68y8ugc0)

[Servidor Web Vulnerable 11](#_heading=h.pq2jkpqnaouz)

[Servidor de Base de Datos Vulnerable 12](#_heading=h.ki02jl81kv8g)

[Servidor FTP Vulnerable 12](#_heading=h.z6mw55ulpxu0)

[Recomendaciones de Seguridad 12](#_heading=h.ju0tftb4w12w)

[Referencias Adicionales 12](#_heading=h.325uxvddnrsa)

# Guía de Trabajos Prácticos

# Herramientas y Configuraciones Iniciales

### Software Requerido

Para abordar los trabajos prácticos, necesitarás instalar las siguientes herramientas de software libre. A continuación, se ofrecen instrucciones de instalación para cada una:

#### Wireshark

* Descripción: Analizador de protocolos utilizado para capturar y analizar el tráfico de red.
* Instalación:
  + Accede a https://www.wireshark.org/download.html y descarga la versión correspondiente a tu sistema operativo.
  + Ejecuta el instalador y sigue los pasos recomendados.

#### GNS3

* Descripción: Simulador de red que permite la combinación de dispositivos virtuales y reales.
* Instalación:
  + Visita https://www.gns3.com/software/download y completa el formulario de registro para descargar.
  + Sigue las instrucciones detalladas en el sitio para instalar GNS3 y sus componentes dependientes.

#### Packet Tracer

* Descripción: Herramienta de simulación de red interactiva para el aprendizaje de redes y para la enseñanza de instrucciones técnicas complejas.
* Instalación:
  + Regístrate o inicia sesión en Cisco NetAcad en https://www.netacad.com/courses/packet-tracer.
  + Descarga e instala Packet Tracer siguiendo las instrucciones proporcionadas.

#### Nmap

* Descripción: Herramienta para exploración de red y auditoría de seguridad.
* Instalación:
  + Visita https://nmap.org/download.html y selecciona la versión adecuada para tu sistema operativo.
  + Descarga e instala Nmap siguiendo las instrucciones.

#### Metasploit Framework

* Descripción: Proyecto de seguridad informática que proporciona información sobre vulnerabilidades de seguridad y ayuda en tests de penetración y desarrollo de firmas IDS.
* Instalación:
  + Accede a https://www.metasploit.com/download y elige la versión gratuita (Metasploit Framework).
  + Sigue las instrucciones para tu sistema operativo. /opt/metasploit-framework/bin/msfconsole

#### TCPdump

* Descripción: Herramienta de línea de comandos para capturar y analizar paquetes de red.
* Instalación:
  + En sistemas basados en Linux, puedes instalarlo mediante el gestor de paquetes de tu distribución (por ejemplo, sudo apt-get install tcpdump en Ubuntu).

## 

# Actividad 1: Captura y Análisis de Tráfico con Wireshark

### 

## Objetivos

* Comprender el flujo de datos en una red y cómo se aplican las capas del Modelo TCP/IP en la transmisión de datos.
* Aprender a identificar y analizar distintos protocolos (TCP, UDP, ICMP) y entender su función en la comunicación de red.

## Pasos Detallados

* Preparación y Configuración:
  + Asegúrese de que Wireshark esté instalado en su sistema. Si no, visite https://www.wireshark.org/download.html para descargar e instalar la última versión.
  + Abra Wireshark y seleccione la interfaz de red por la que desea capturar tráfico. Esto puede ser una conexión Ethernet o Wi-Fi dependiendo de su configuración de red.
* Inicio de la Captura de Paquetes:
  + Haga clic en el botón "Start Capturing Packets" para comenzar la captura en la interfaz seleccionada.
  + Realice actividades de red específicas para generar tráfico. Por ejemplo, navegue por diferentes sitios web, envíe correos electrónicos, realice conexiones SSH o FTP, entre otros.
* Filtrado y Análisis de Tráfico:
  + Utilice filtros en Wireshark para aislar tipos específicos de tráfico. Por ejemplo, tcp, udp, o icmp para ver solo paquetes que usen estos protocolos.
  + Examine los detalles de los paquetes capturados, incluyendo las cabeceras de los protocolos en las distintas capas (Enlace de datos, Red, Transporte, Aplicación) y los datos transportados.
* Identificación de Patrones de Tráfico:
  + Identifique patrones de tráfico típicos, como un handshake de TCP, consultas DNS, o el flujo de una sesión HTTP.
  + Analice cualquier anomalía o patrón inusual que podría indicar problemas de red o actividad sospechosa.
* Documentación y Reporte:
  + Documente sus observaciones, incluyendo capturas de pantalla de paquetes representativos o interesantes.
  + Escriba un breve informe que describa los protocolos analizados, las observaciones hechas durante la captura, y cualquier conclusión o recomendación resultante.

## Referencias Teóricas

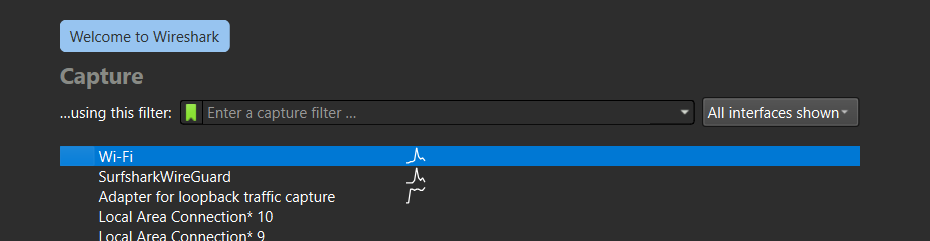
* "Wireshark Network Analysis (Second Edition): The Official Wireshark Certified Network Analyst Study Guide" por Laura Chappell. Este libro es una excelente fuente de aprendizaje para profundizar en el análisis de redes con Wireshark.
* "TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference" por Charles M. Kozierok. Proporciona una visión detallada del Modelo TCP/IP, protocolos y conceptos fundamentales en redes.
* RFC 791 (Protocolo IP), RFC 793 (TCP), y RFC 768 (UDP): Documentos técnicos que definen los protocolos IP, TCP y UDP, respectivamente. Disponibles en https://www.ietf.org/rfc.html.

## Pasos Adicionales para la Exploración

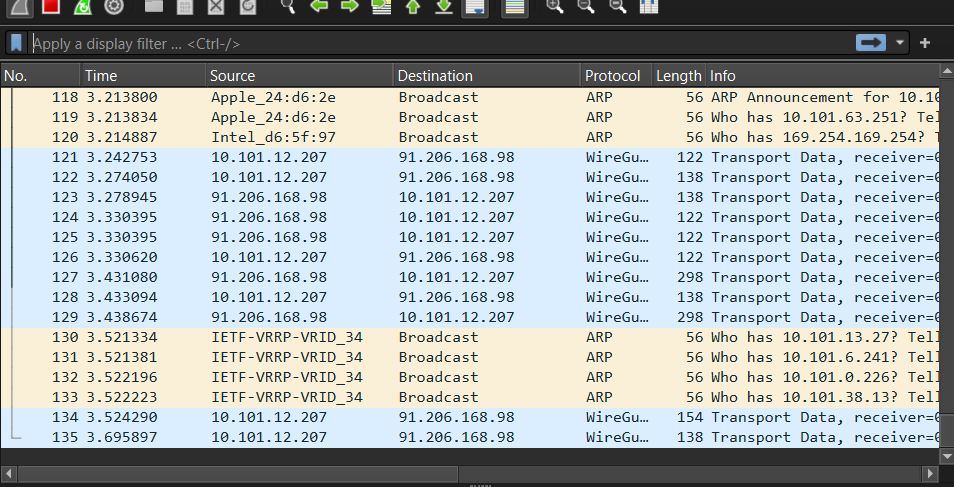
* Análisis de Streams de Conversación: Utilice la funcionalidad "Follow TCP Stream" en Wireshark para visualizar el flujo completo de una conversación TCP. Esto es útil para entender cómo se intercambian los datos a lo largo de una sesión.
* Uso de Estadísticas: Explore las herramientas estadísticas de Wireshark, como "Protocol Hierarchy" y "Endpoints", para obtener una visión general del tráfico de red y los participantes más activos.

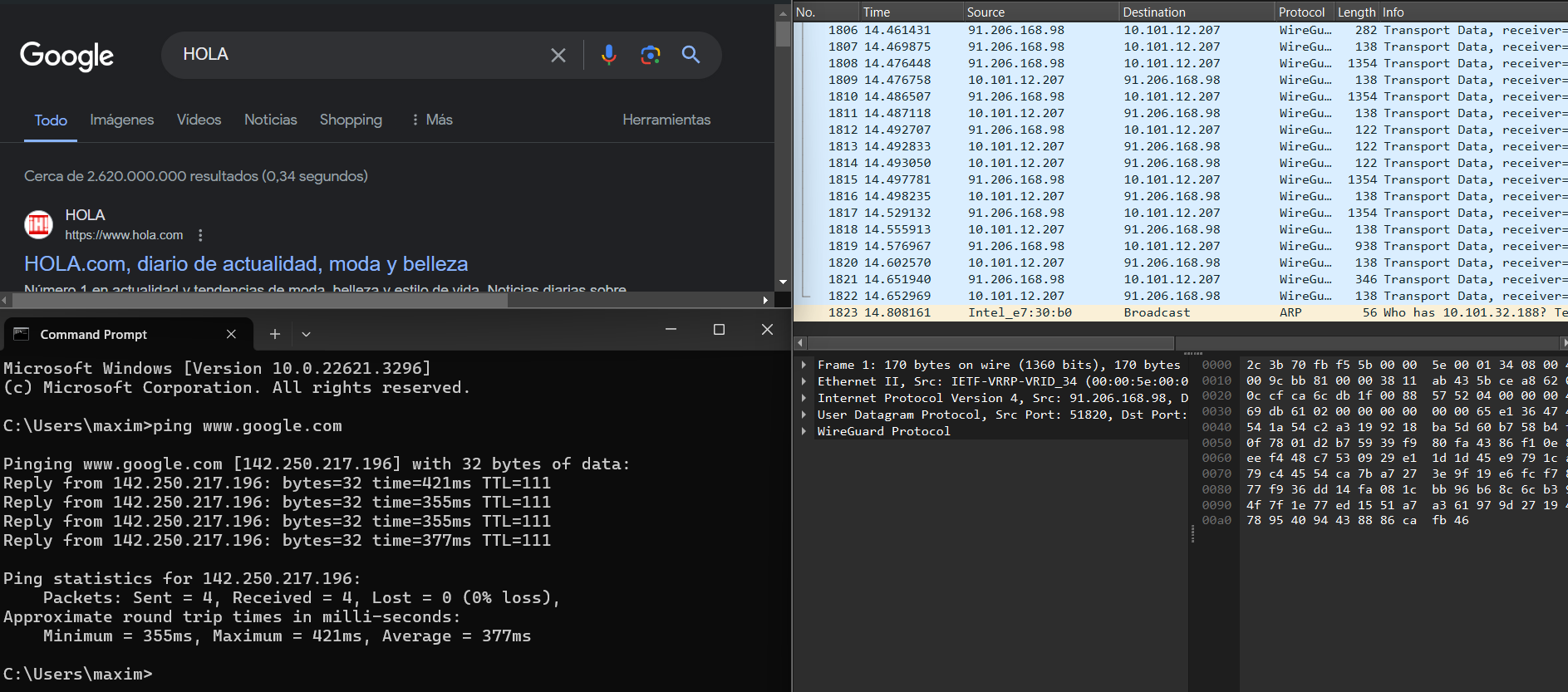
# Actividad 1: Resolución

**Preparación y Configuración:**

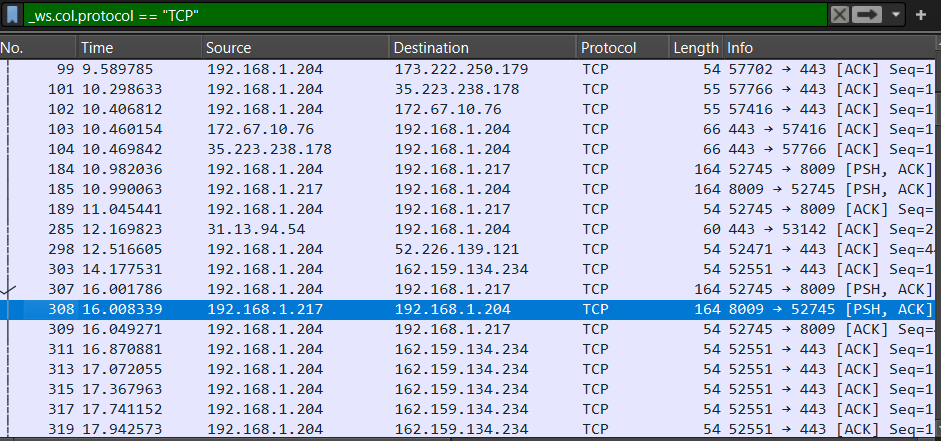


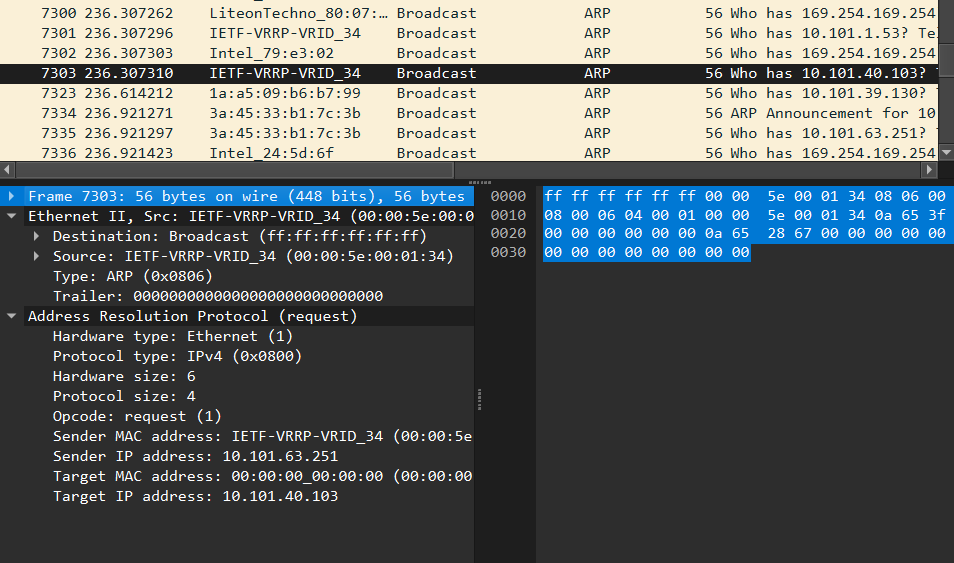
**Inicio de la Captura de Paquetes:**



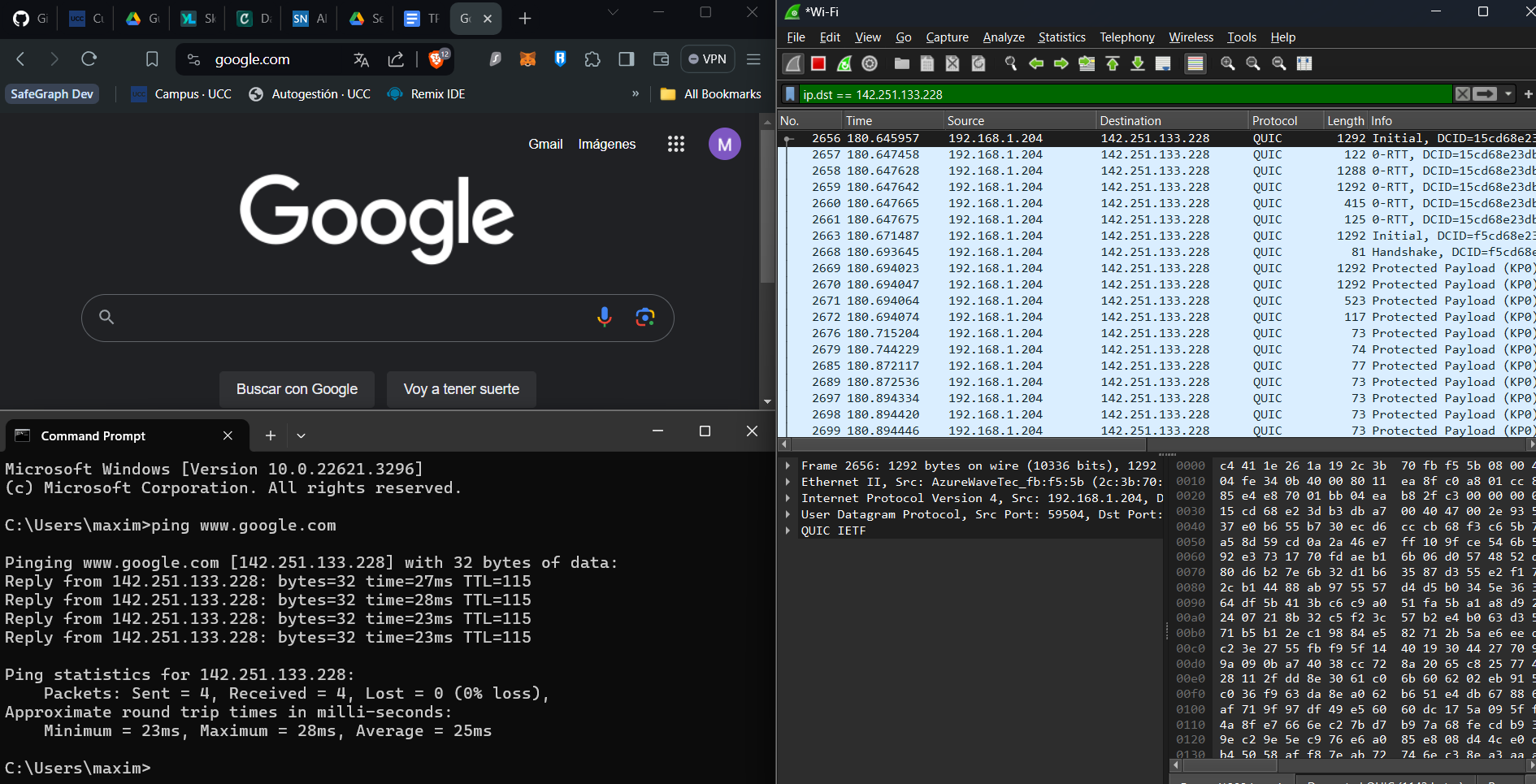


**Filtrado y Análisis de Tráfico:**





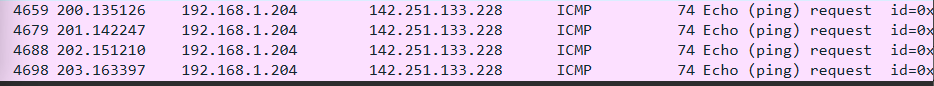
**Identificación de Patrones de Tráfico:**



Se puede visualizar la llamada inicial para el handshake HTTP después de resolver el dominio ‘[www.google.com](http://www.google.com)’ en el navegador web.

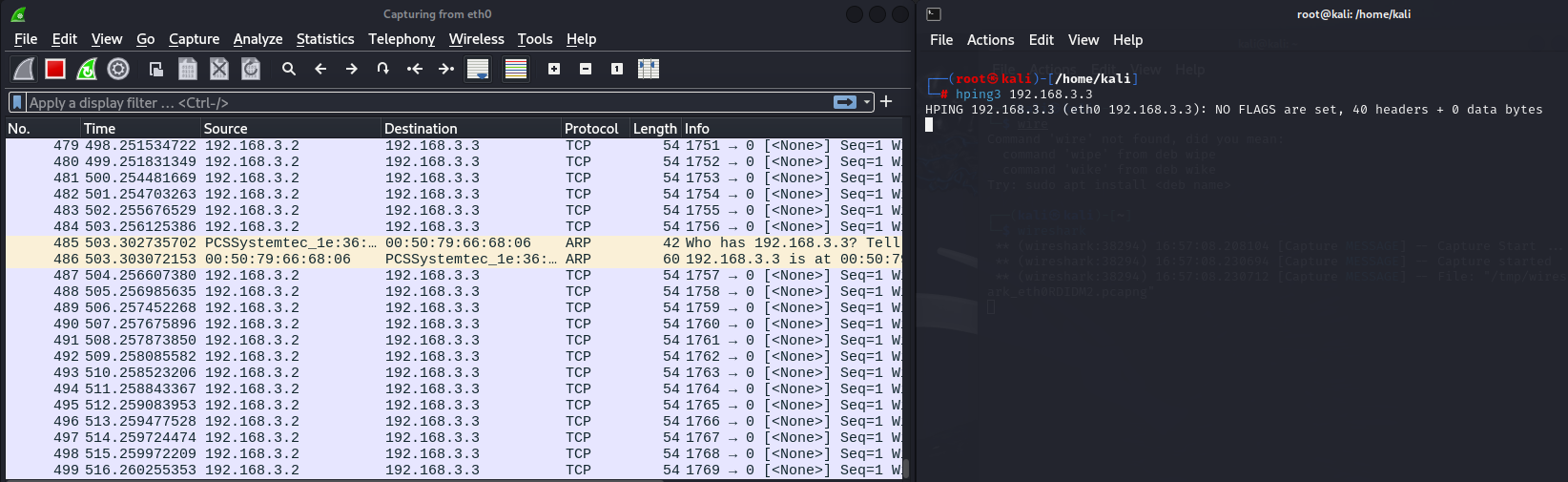
Se realizó un ping al dominio para verificar la IP de destino para así poder filtrar en los registros de Wireshark.

El trafico generado por el ping al dominio ‘[www.google.com](http://www.google.com)’ desde la consola CMD también quedó registrado en Wireshark bajo el protocolo ICMP de ‘echo’ a la red.



Son esas cuatros lineas. Corresponden a los cuatro pings que se hicieron con el comando ‘ping www.google.com’.

Abajo podemos ver un caso tipo de DoS siendo registrado por Wireshark.



## 

## 

# 

# Actividad 2: Diseño y Configuración de Redes con GNS3

## Objetivos

* Diseñar una red que incorpore múltiples subredes y aprenda a configurar dispositivos de red para facilitar la comunicación entre ellas.
* Comprender y aplicar conceptos de enrutamiento estático y dinámico para gestionar el tráfico de red eficazmente.

## Pasos Detallados

* Preparación y Entorno de GNS3:
  + Instale GNS3 si aún no lo ha hecho. Las instrucciones de instalación se pueden encontrar en https://www.gns3.com/software/download.
  + Familiaricese con la interfaz de GNS3. Consulte la documentación oficial y tutoriales para una introducción básica.
* Diseño de la Red:
  + Utilice GNS3 para dibujar un esquema de red que incluya, al menos, tres subredes, cada una conectada por routers. Asegúrese de incluir al menos un switch en cada subred y múltiples hosts (máquinas virtuales o emuladas).
  + Asigne rango de direcciones IP distintas para cada subred para simular un entorno de red real.
* Configuración de Dispositivos:
  + Configure las interfaces de red de los routers con direcciones IP estáticas adecuadas para cada subred.
  + Configure los dispositivos finales (PCs o servidores emulados) con direcciones IP estáticas o dinámicas, asegurándose de que cada dispositivo tenga configurada la puerta de enlace predeterminada correctamente.
* Implementación de Enrutamiento:
  + Implemente enrutamiento estático definiendo rutas específicas en los routers para permitir la comunicación entre subredes.
  + Para una experiencia más avanzada, configure un protocolo de enrutamiento dinámico como RIP, OSPF o EIGRP en los routers para automatizar el proceso de descubrimiento de rutas.
* Pruebas y Validación:
  + Utilice herramientas de línea de comandos como ping y tracert (o traceroute en Linux) para verificar la conectividad entre dispositivos en diferentes subredes.
  + Analice el comportamiento del tráfico utilizando la captura de paquetes integrada en GNS3 para entender cómo se enrutan los paquetes a través de la red.
* Documentación y Análisis:
  + Documente la configuración de su red, incluyendo un diagrama de red, tablas de enrutamiento de cada router y la configuración IP de cada dispositivo.
  + Reflexione sobre cómo el enrutamiento estático o dinámico afecta la eficiencia y la escalabilidad de la red.

## Referencias Teóricas

* "Computer Networking: A Top-Down Approach" por James Kurose y Keith Ross. Este libro proporciona una excelente introducción a los fundamentos de las redes, incluyendo la configuración de subredes y enrutamiento.
* "Routing TCP/IP, Volume 1" por Jeff Doyle. Profundiza en el enrutamiento dentro de las redes IP, cubriendo tanto el enrutamiento estático como dinámico, y es una lectura esencial para comprender los protocolos de enrutamiento.
* RFC 791 (IP), RFC 1058 (RIP), RFC 2328 (OSPF), y RFC 2453 (RIP v2): Proporcionan detalles técnicos de los protocolos de red y enrutamiento.

## Pasos Adicionales para la Exploración

* Seguridad de Red: Introduzca configuraciones de firewall en los routers para limitar el acceso entre subredes. Experimente con reglas ACL (listas de control de acceso) para filtrar tráfico específico.
* Simulación de Fallas: Intencionalmente configure rutas incorrectas o desconecte interfaces de red para simular fallos. Analice cómo la red responde a estas fallas y cómo los protocolos de enrutamiento dinámico se adaptan a cambios en la topología de la red.

# Actividad 2: Resolución

**Diseño de la red:**

Se diseñaron 3 subredes con los siguientes rangos de ips:

Subred 1:

Submask Class C: 255.255.255.0

IP Address: 192.168.1.X

IP Gateway (Switch 1): 192.168.1.1

Subred 2:

Submask Class B: 255.255.0.0

IP Addresses: 172.16.0.X

IP Gateway (Switch 2): 172.16.0.

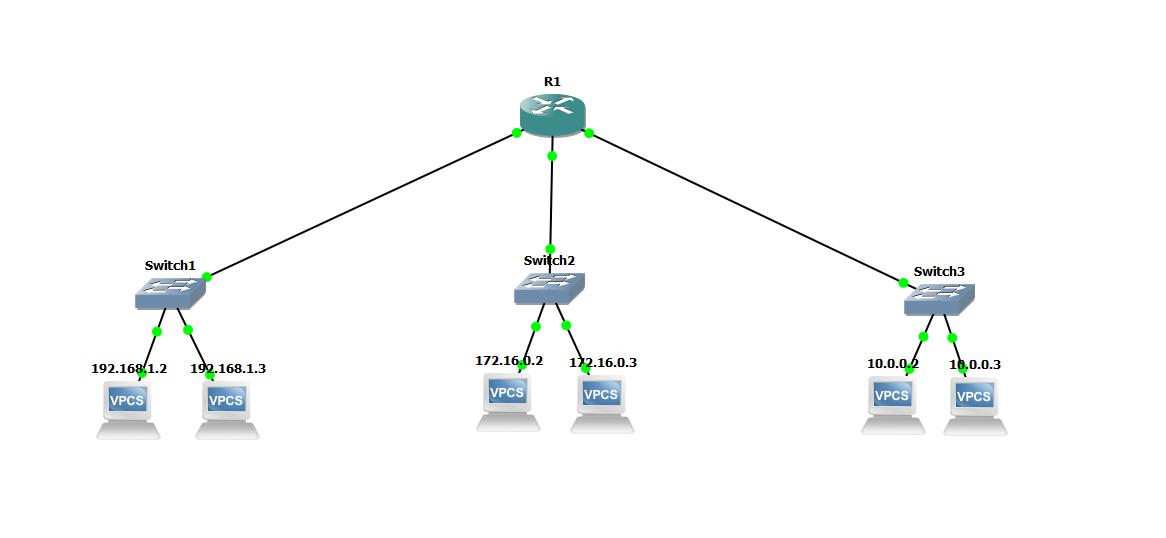
Subred 2:

Submask Class B: 255.0.0.0

IP Addresses: 10.0.0.X

IP Gateway (Switch 3): 10.0.0.1

Diagrama:

**Configuración de los dispositivos:**

Para configurar los hosts se accedió a la consola de cada dispositivo final y con el comando ‘ip’ se seteo la IP address y IP gateway:

#ip 192.168.1.2/24 192.168.1.1 .

‘192.168.1.2’ es la dirección IP que se le asignó, ‘/24’ indica la cantidad de bits de enmascaramiento (Clase C) y ‘192.168.1.1’ indica la ip gateway

**Implementación del enrutamiento:**

Para lograr esto se accedió a la consola del router y se ejecutaron los siguientes comandos para configurar cada interfaz para cada subred:

#conf t .

(conf)#interface FastEthernet0/0 .

(conf-if)#ip address 192.168.1.1 255.255.255.0 .

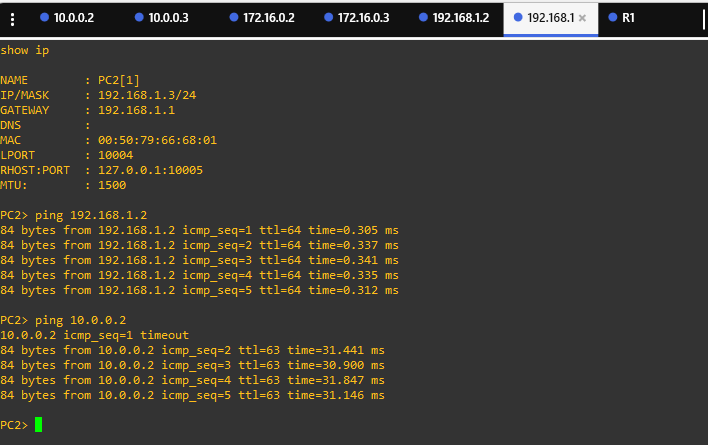
(conf-if)#no shutdown .

(conf-if)#exit .

Se repiten estos comandos para las otras subredes

**Pruebas y validación:**

Aquí se puede ver un ping entre dispositivos de las subredes



## 

# Actividad 3: Simulación de Ataques y Defensas en Red

## Objetivos

* Comprender y aplicar técnicas de ataque comunes en un entorno controlado y ético para identificar vulnerabilidades en la red.
* Desarrollar e implementar estrategias de defensa efectivas para mitigar y proteger la red contra ataques simulados.

## Pasos Detallados

* Preparación del Entorno de Ataque:
  + Configure una red de laboratorio en GNS3 que incluya varios dispositivos y configuraciones de red que reflejen un entorno empresarial realista. Esto debe incluir servidores, estaciones de trabajo y dispositivos de red como routers y switches.
  + Asegúrese de que todas las actividades de ataque se realicen en este entorno aislado para evitar impactos no intencionados.
* Simulación de Ataques:
  + Ataque de ARP Spoofing: Utilice herramientas como Ettercap o Arpsoof para realizar un ataque de ARP Spoofing. Este ataque permite al atacante interceptar la comunicación entre dos hosts en la misma red local.
  + Denegación de Servicio (DoS): Implemente un ataque DoS simple utilizando herramientas como hping3 para enviar una cantidad abrumadora de tráfico a un servidor objetivo, intentando sobrecargar sus recursos.
* Implementación de Medidas Defensivas:
  + Defensa contra ARP Spoofing: Configure características de seguridad en switches, como Dynamic ARP Inspection (DAI), para bloquear respuestas ARP no válidas.
  + Mitigación de Ataques DoS: Implemente reglas de control de acceso y tarifado de tráfico en los dispositivos de red para limitar la tasa de paquetes aceptados de fuentes no confiables.
* Evaluación de la Eficacia de las Defensas:
  + Realice nuevamente los ataques después de implementar las medidas defensivas para evaluar su eficacia.
  + Utilice herramientas de monitoreo y análisis de tráfico, como Wireshark, para observar la diferencia en el comportamiento de la red y la efectividad de las defensas aplicadas.
* Reporte y Análisis:
  + Documente cada paso del proceso de ataque y defensa, incluyendo configuraciones específicas utilizadas, herramientas y técnicas aplicadas, y las observaciones del impacto y efectividad de las medidas defensivas.
  + Reflexione sobre cómo cada ataque afectó la red y discuta las lecciones aprendidas sobre la importancia de las estrategias de defensa en la seguridad de la red.

## Referencias Teóricas

* "The Basics of Hacking and Penetration Testing" por Patrick Engebretson: Proporciona una introducción a las técnicas de hacking ético y pruebas de penetración, incluyendo ataques de ARP Spoofing y DoS.
* "Network Security Essentials" por William Stallings: Ofrece una visión general de los conceptos de seguridad de red, incluyendo defensas contra diversos tipos de ataques cibernéticos.
* RFC 826 (ARP) y RFC 2827 (Filtrado de Tráfico Inbound): Proporcionan información técnica sobre el protocolo ARP y las mejores prácticas para el filtrado de tráfico, respectivamente, que son cruciales para entender y defenderse contra ataques como ARP Spoofing y DoS.

## Pasos Adicionales para la Exploración

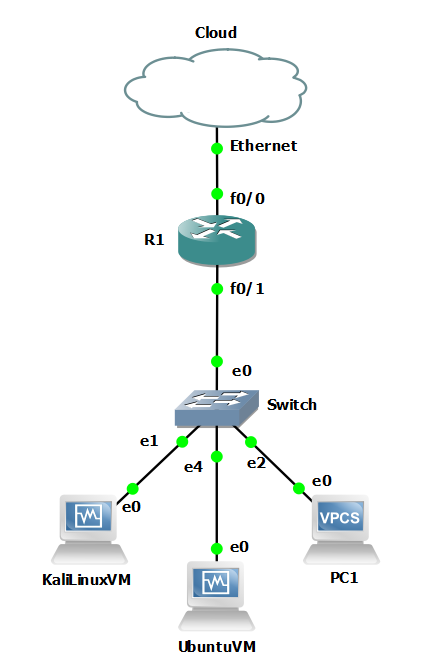
* Simulación de Ataques de Ingeniería Social: Aunque más difíciles de simular en un entorno de laboratorio, discuta los conceptos y técnicas de ingeniería social. Reflexione sobre cómo la concienciación y formación de los usuarios puede ser una defensa crucial.
* Uso de IDS/IPS: Explore la configuración y el despliegue de Sistemas de Detección/Prevención de Intrusos (IDS/IPS) dentro de la red de laboratorio para detectar y prevenir ataques automatizados y sofisticados.

## 

# 

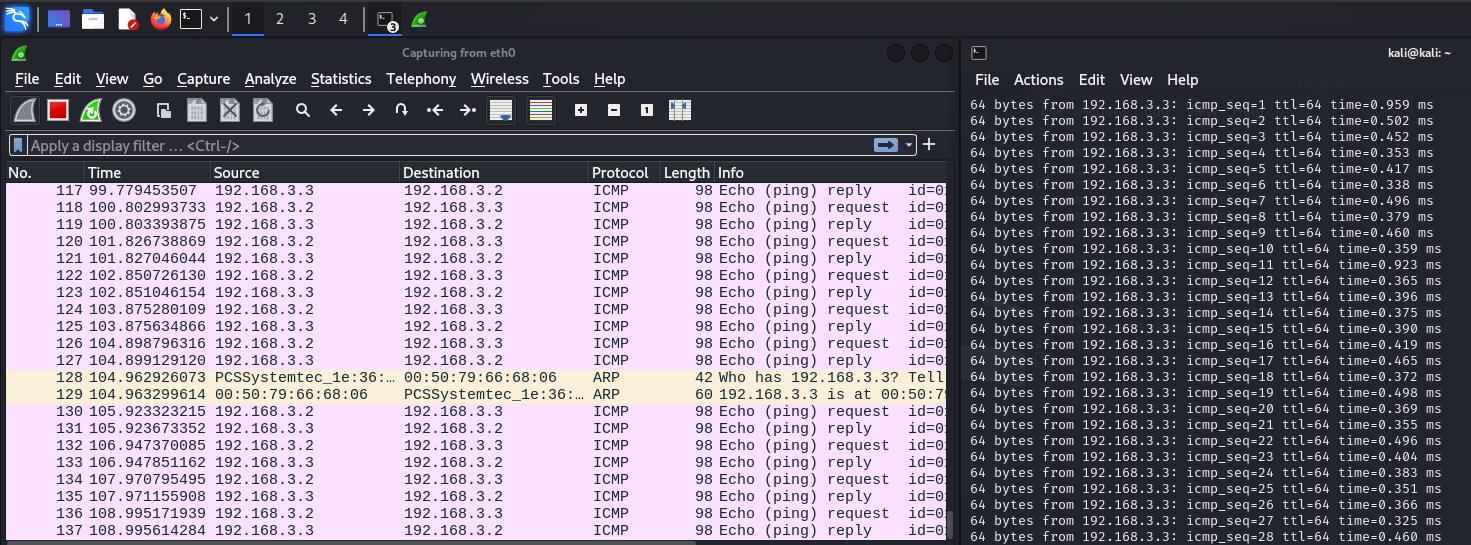
# Actividad 3: Resolución

Esquema de red para realizar el ciberataque:

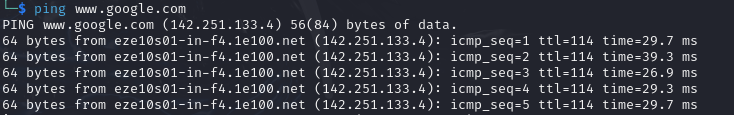
**PC1 (192.168.3.3 - NAT 192.168.1.102)** será utilizada como target del Ataque DoS, mientras que **UbuntuVM (192.168.3.4 - NAT 192.168.1.103)** será utilizado como target del Ataque ARP Spoofing o Poisoning. **KaliLinuxVM (192.168.3.2 - NAT 192.168.1.101)** serála máquina atacante.

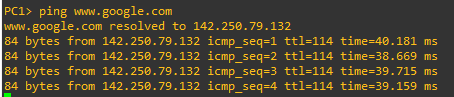
## DoS Attack

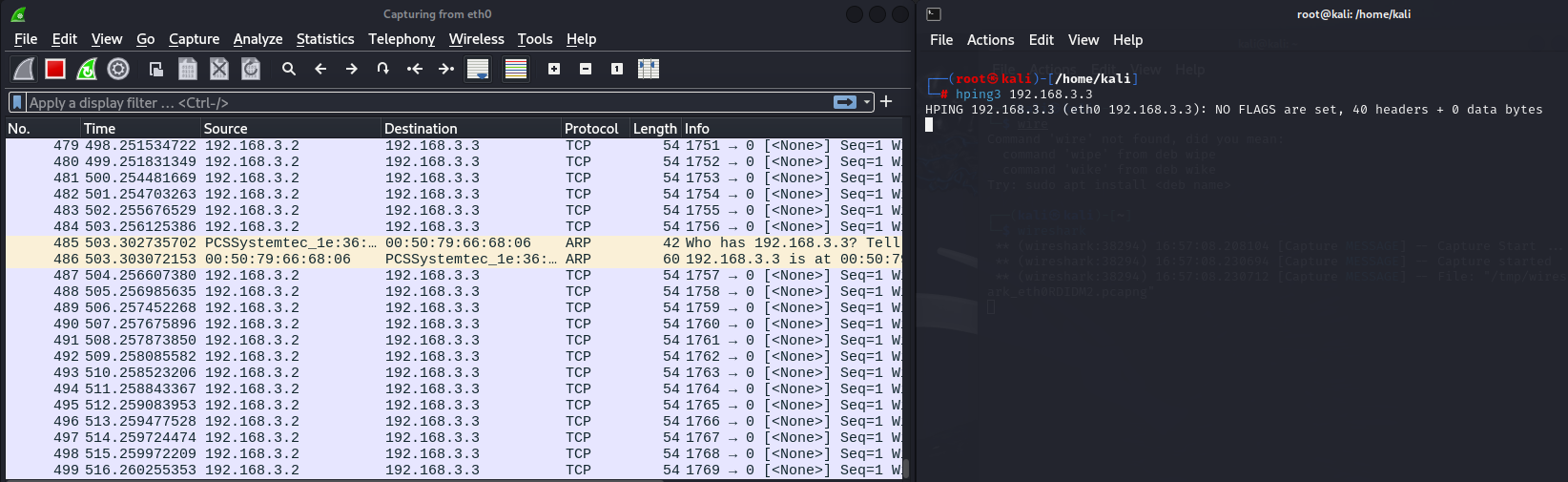
Comprobando que las máquinas se puedan comunicar entre sí enviando una señal de ping desde la Kali Linux VM a la VPCS de GNS3.



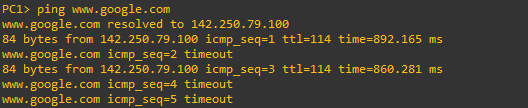
Ambas computadoras estan enrutadas para disponer de conexion a internet y ademas tienen configurada una direccion de DNS para poder resolver los dominios:





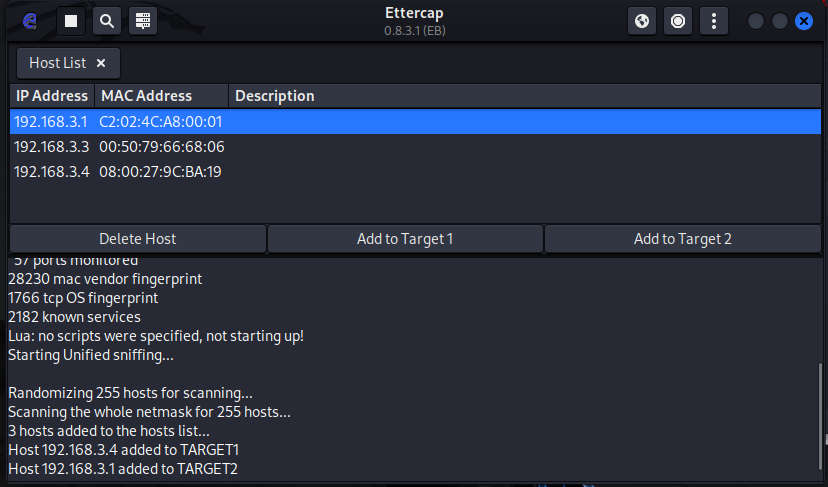
Luego de comprobar que la computadora víctima del ataque dispone de conexión a internet, procederemos a hacer un Ataque DoS (utilizando **hping3**) y posteriormente verificaremos su efectividad realizando el mismo ping a ‘[www.google.com](http://www.google.com)’ esperando que no haya respuesta.  


Podemos notar como tras realizar el ataque la latencia de recepción de paquetes aumenta significativamente en algunos casos mientras que en otros directamente se pierden por completo.



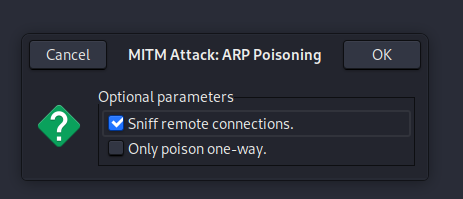
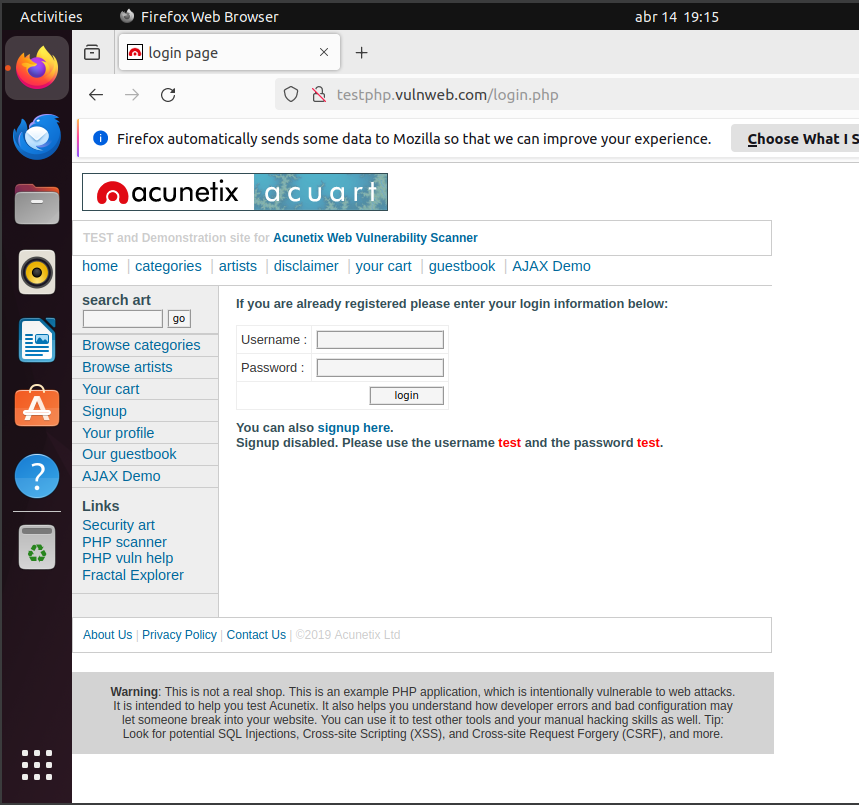
## ARP Spoofing Attack

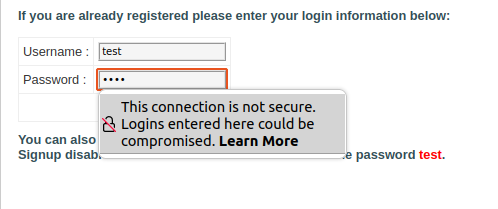
Configurando Ettercap para un Ataque de Man In The Middle (MITM) ARP Spoofing.

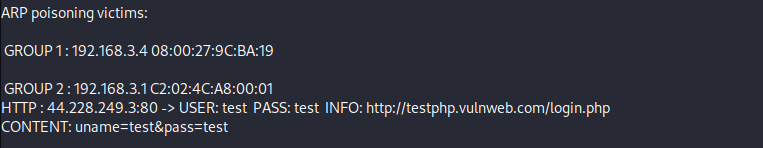


Vamos a tratar de capturar las credenciales de inicio de sesión enviadas desde la máquina UbuntuVM a través de una pantalla de login de la pagina ‘<http://www.vulnweb.com/>’. El protocolo debe ser HTTP en este caso para facilitar el ejercicio, ya que de ser HTTPS,

la información capturada estaría encriptada.

Iniciamos el Ataque ARP Poisoning (ARP Spoofing) desde Ettercap hacia los targets configurados.  


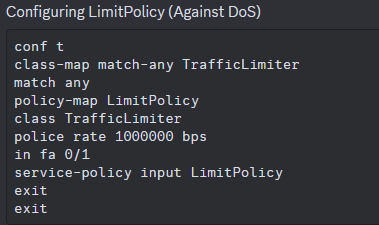
Ingresamos las credenciales de login desde la máquina Ubuntu, y posteriormente verificamos si han sido capturadas por Ettercap.  




**Ataque exitoso! ¡EL PRÓXIMO ES USTED!**

## Defensa contra los Ciberataques

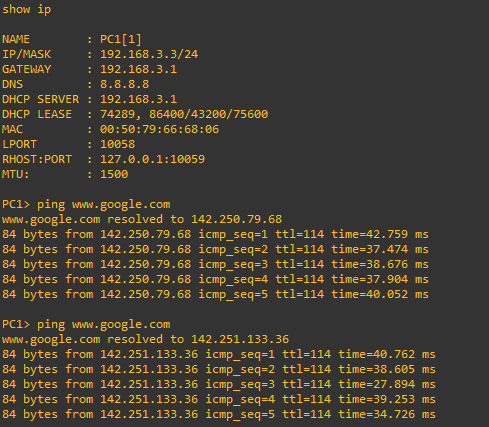
Configuramos una Política de RateLimit para el puerto del Router que contiene la subred a la que están conectadas el atacante y las víctimas, para limitar el tráfico que los dispositivos pueden enviar y recibir buscando impedir ataques de tipo DoS.



Luego de configurarla realizaremos el ataque desde **KaliLinuxVM** a la **PC1** de nuevo para ver si la defensa surte efecto.

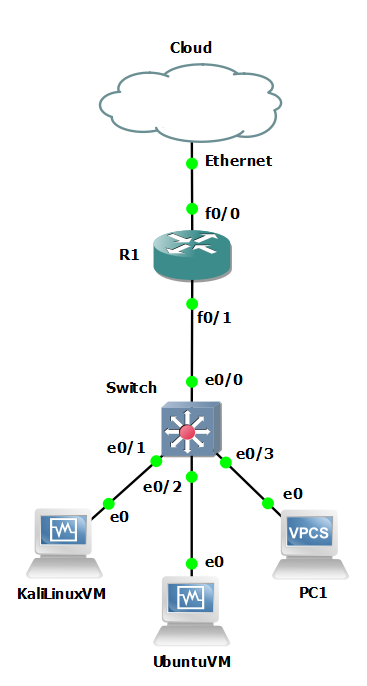


# 



El primer ping fue realizado antes de iniciar el Ataque DoS y el segundo después de haberlo iniciado, como podemos ver la defensa logró bloquear exitosamente el ataque.

Para la defensa contra ARP Spoofing debimos actualizar el esquema cambiando el switch por uno CISCO compatible con consola para poder levantar la barrera protectora.   
El nuevo esquema quedo asi:



# 

Para proteger la red contra Ataques MITM ARP Spoofing haremos uso del concepto Network Segmentation, que consiste en agrupar máquinas críticas con funcionalidades similares en distintas VLAN o canales aislados entre sí.



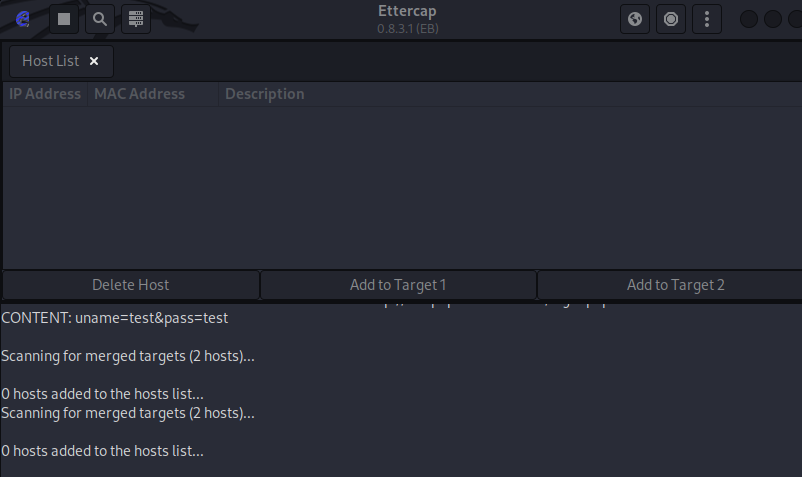




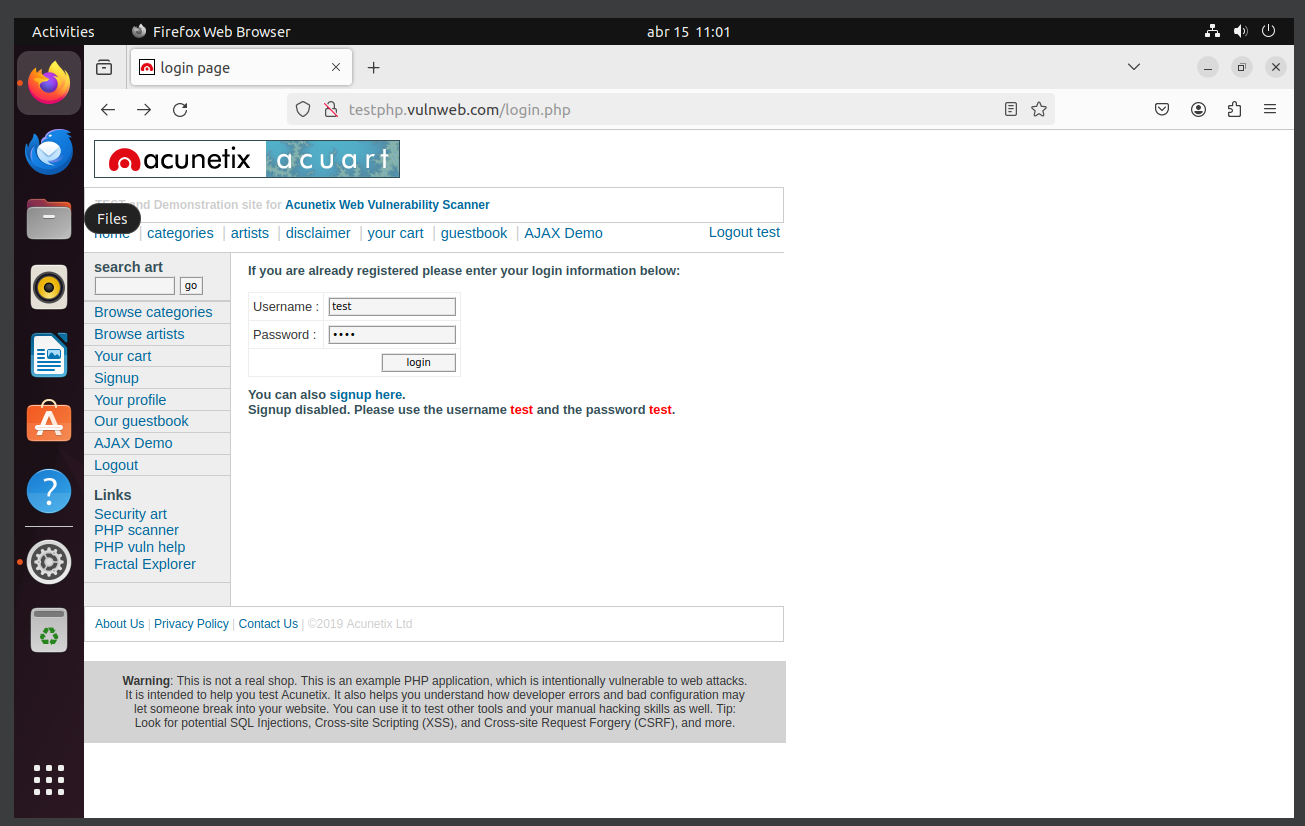


**Nota**: La máquina atacante se encuentra en la VLAN 10, mientras que la máquina victima se encuentra en la VLAN 20.

Luego de configurar en que VLAN se encuentra cada puerto del Switch CISCO intentaremos realizar el mismo Ataque MITM ARP Spoofing de antes y probaremos si tras levantar las barreras defensivas contra este tipo de ataque, el ataque sigue surtiendo efecto o no.



Tras realizar un escaneo de hosts en la red, podemos ver que la barrera ha funcionado exitosamente, ya que la máquina atacante ni siquiera puede visualizar los dispositivos sensibles a pesar de estar conectada a la misma red física que los anteriores mencionados.



# Trabajo Integrador: Auditoría de Seguridad de una Red

### Objetivo

Realizar una auditoría de seguridad completa de una red simulada, identificando vulnerabilidades y proponiendo medidas correctivas.

### Descripción del Proyecto

* Diseño de la Red: Crear una red en GNS3 que simule una pequeña empresa, con servidores, estaciones de trabajo y dispositivos de red.
* Análisis de Vulnerabilidades:
  + Utilizar Nmap para identificar servicios expuestos y posibles puntos de entrada.
  + Emplear herramientas como Metasploit para evaluar las vulnerabilidades encontradas.
* Simulación de Ataques:
  + Realizar ataques controlados, con permiso, para determinar el impacto potencial de las vulnerabilidades identificadas.
* Desarrollo de un Plan de Mitigación:
  + Basándose en los hallazgos, proponer un plan de acción para mitigar las vulnerabilidades identificadas.
* Reporte Final:
  + Elaborar un informe detallado que incluya la metodología utilizada, los hallazgos, el análisis de impacto y las recomendaciones de mitigación.

## Referencias Bibliográficas

* "Applied Cryptography" por Bruce Schneier (Wiley, 2015) - Capítulos 1-5.
* "Practical UNIX and Internet Security" por Simson Garfinkel y Gene Spafford (O'Reilly Media, 2017) - Capítulos 10-15.
* "OWASP Top Ten" (2022) - Lista completa y descripciones disponibles en https://owasp.org/www-project-top-ten/.
* "Practical Malware Analysis" por Michael Sikorski y Andrew Honig (No Starch Press, 2012) - Capítulos 1, 2, y 7.
* "Cloud Security: A Comprehensive Guide to Secure Cloud Computing" por Ronald L. Krutz y Russell Dean Vines (Wiley, 2010) - Capítulos 3 y 4.

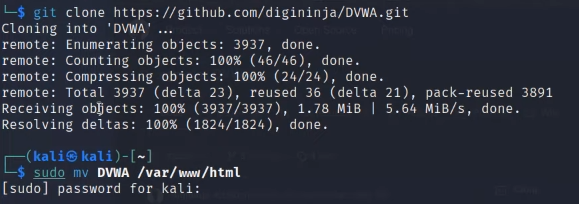
Además, se recomienda revisar los recursos en línea proporcionados en las instrucciones de instalación de cada herramienta para guías específicas y documentación adicional.

# Trabajo Integrador

# Empezamos levantando 2 máquinas virtuales con Kali Linux. Kali-1 estará levantando un servicio web con algunas vulnerabilidades. Kali-2 será la encargada de explotar las vulnerabilidades de Kali-1.

# 

# Procedemos levantar y configurar el servicio DVWA(DAMN VULNERABLE WEB APPLICATION) desde Kali-1.

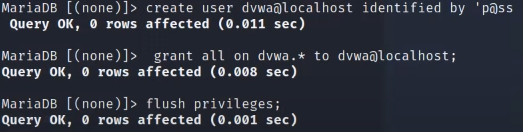


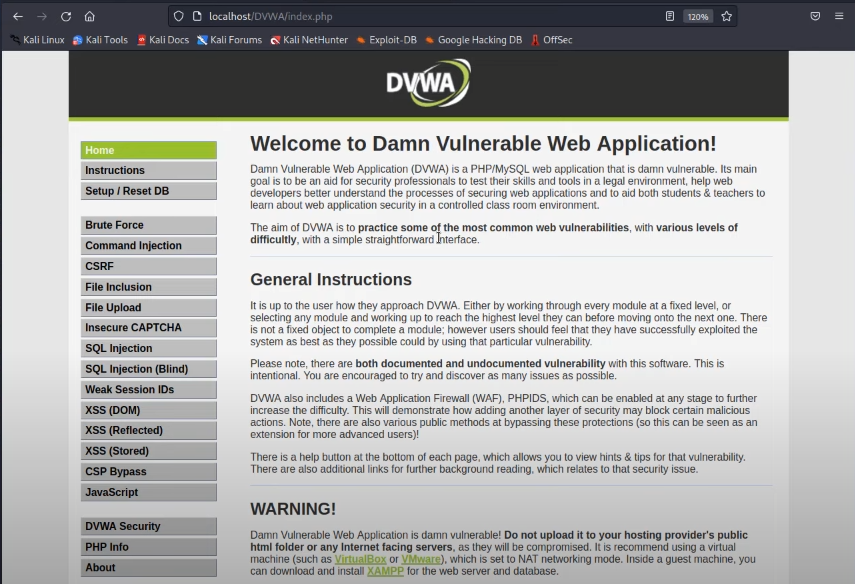
# 



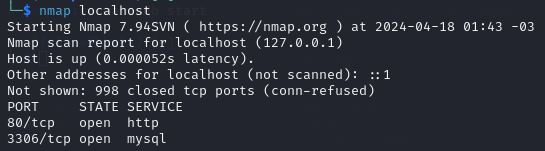








Chequeamos con Nmap los servicios levantados.



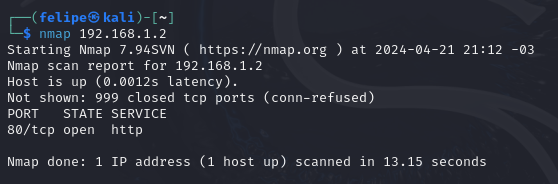
Seteamos ip en la máquina virtual Kali-1:



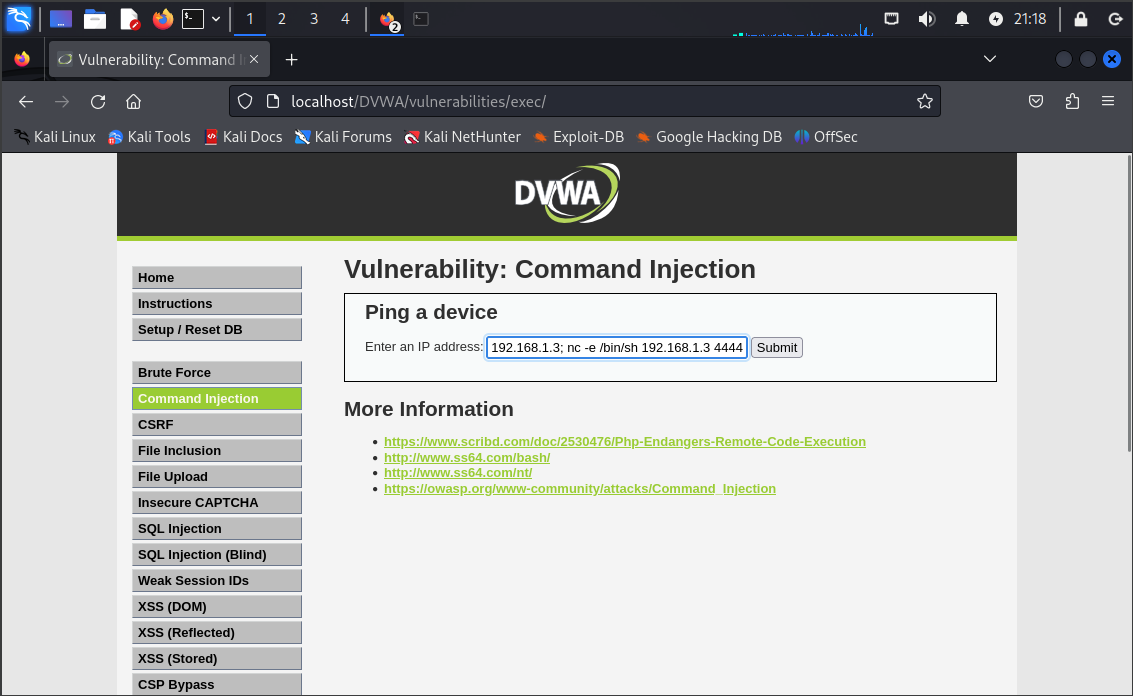
Seteamos ip de la máquina virtual Kali-2:



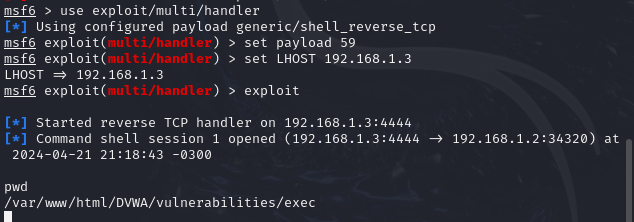
Chequeamos con Nmap desde Kali-2 los servicios levantados en Kali-1.



Esta página ofrece la funcionalidad de realizar un ping a una dirección IP específica. Sin embargo, el input destinado a la selección de la IP para realizar el ping permite la inserción de comandos. Dado que esta función utiliza directamente el comando ping en la consola, existe una clara vulnerabilidad de "Command Injection".



Desde Kali-2 usando Metasploit podremos acceder a la consola de Kali-1:



**Plan de mitigación**

Para mitigar esta vulnerabilidad, primero necesitas entender que el problema radica en la forma en que el servicio web ejecuta comandos en el sistema subyacente. Esto es extremadamente arriesgado, ya que permite a los usuarios enviar comandos arbitrarios al sistema operativo que aloja el servicio, lo que puede conducir a una ejecución de código remoto no autorizada.

Aquí hay un plan de mitigación para abordar esta vulnerabilidad:

* Validación de entrada: Implementa una estricta validación de entrada en el servicio web para filtrar y rechazar cualquier entrada maliciosa o inesperada. Esto puede incluir la verificación de caracteres especiales, longitud de entrada, formato esperado, etc.
* Desactivar comandos de shell: Configura el servicio web para no ejecutar comandos del sistema directamente. Esto implica cambiar el enfoque del diseño para que el servicio no utilice la consola del sistema operativo para realizar acciones.
* Utilizar librerías seguras: Si es necesario ejecutar comandos del sistema, utiliza bibliotecas específicas del lenguaje de programación que ofrezcan funcionalidades seguras para ejecutar comandos de forma controlada, como subprocess en Python.
* Aislamiento del entorno: Ejecuta el servicio web en un entorno aislado y restringido, como contenedores Docker, máquinas virtuales o chroot jails. Esto limita el impacto de posibles ataques al restringir el acceso a recursos críticos del sistema.
* Principio de menor privilegio: Configura el servicio web para que se ejecute con los privilegios mínimos necesarios para su funcionamiento. Esto reduce el impacto potencial de cualquier compromiso del servicio.

Implementando estos pasos, puedes reducir significativamente la exposición a esta vulnerabilidad y fortalecer la seguridad de tu servicio web.

# APÉNDICE 1- Generación de Servicios con Vulnerabilidades Específicas

Como parte del trabajo práctico integrador, se requiere establecer un entorno que simule servicios con vulnerabilidades conocidas. Esto se hará configurando versiones específicas de software que sean conocidas por tener fallos de seguridad que los estudiantes podrán explorar y mitigar.

## Configuración de Servicios Vulnerables

### 

### Servidor Web Vulnerable

* Objetivo: Configurar un servidor web con vulnerabilidades específicas que puedan ser explotadas.
* Herramientas y Software:
  + Docker: Utilizará imágenes de Docker para desplegar rápidamente entornos con vulnerabilidades conocidas.
  + OWASP WebGoat o DVWA (Damn Vulnerable Web Application): Son aplicaciones web diseñadas para enseñar y practicar técnicas de seguridad web.
* Configuración:
  + Instalación de Docker: Asegúrese de que Docker esté instalado en su sistema. Si no, puede descargarlo desde https://www.docker.com/products/docker-desktop.
  + Despliegue de DVWA:
    - Ejecute el comando docker pull vulnerables/web-dvwa para descargar la imagen de Docker de DVWA.
    - Inicie la instancia de DVWA con el comando docker run -d -p 80:80 vulnerables/web-dvwa. Esto hará que DVWA esté accesible en el puerto 80 de su máquina local.

### Servidor de Base de Datos Vulnerable

* Objetivo: Instalar una versión antigua de MySQL o MongoDB que contenga vulnerabilidades específicas.
* Configuración:
  + Utilice contenedores de Docker para instalar versiones específicas que contengan vulnerabilidades conocidas, como MySQL 5.5 o MongoDB 2.6.
  + Ejemplo para MySQL:
    - Ejecute docker pull mysql:5.5 para obtener una imagen vulnerable de MySQL.
    - Inicie el contenedor con docker run -d -p 3306:3306 --name mysql-vuln mysql:5.5, asegurándose de configurar cualquier variable de entorno necesaria para su inicialización.

### Servidor FTP Vulnerable

* Objetivo: Desplegar un servidor FTP que contenga vulnerabilidades para practicar la explotación y mitigación.
* Configuración:
  + Busque una imagen de Docker o configure manualmente una versión vulnerable de vsFTPd, por ejemplo, la versión 2.3.4 que es conocida por tener una vulnerabilidad explotable.
  + Ejemplo con Docker: Si existe una imagen disponible de vsFTPd 2.3.4, utilícela; de lo contrario, puede configurar un entorno virtualizado y descargar específicamente esta versión para su configuración.

## 

## Recomendaciones de Seguridad

* Aislamiento: Asegúrese de que estos entornos vulnerables estén aislados y solo accesibles en un entorno de laboratorio controlado. Nunca exponga estos servicios a redes públicas o a Internet.
* Monitorización: Utilice herramientas como Wireshark y el registro de eventos del sistema para monitorear el tráfico y las acciones realizadas durante las sesiones de prueba. Esto es esencial para el aprendizaje y la comprensión de las vulnerabilidades.
* Ética: Recuerde la importancia de la ética en la seguridad informática. Estos entornos están destinados únicamente para fines educativos y cualquier conocimiento adquirido debe usarse de manera responsable.

## Referencias Adicionales

* OWASP WebGoat:<https://github.com/WebGoat/WebGoat>
* DVWA - Damn Vulnerable Web Application:<https://dvwa.co.uk/>
* Docker Hub: Para buscar imágenes de software específico, visite https://hub.docker.com/.