

Análisis de Vulnerabilidades y Vectores de Ataque

Club de Programación - UNLP

Agenda Técnica de la Sesión

01

Metodologías de Pentesting y Reconocimiento de Red

Exploraremos los frameworks estándar de la industria para pentesting, incluyendo PTES y OSSTMM. Aprenderemos por qué seguir una metodología estructurada es crucial para obtener resultados profesionales y consistentes.

Comenzaremos con técnicas de network discovery para mapear el entorno objetivo.

02

Enumeración Sistemática de Servicios de Red y Web

Profundizaremos en las técnicas de escaneo de puertos y detección de servicios utilizando herramientas como nmap. Analizaremos cómo interpretar los resultados para identificar vectores de ataque potenciales y construir un perfil completo de la superficie de ataque.

03

Correlación de Vulnerabilidades y Formulación de Vectores de Ataque

Conectaremos los hallazgos del reconocimiento con vulnerabilidades conocidas. Aprenderemos a priorizar objetivos y desarrollar estrategias de explotación basadas en la información recopilada durante las fases anteriores.

Frameworks de Pruebas de Penetración

La Importancia de una Metodología Estructurada

PTES (Penetration Testing Execution Standard)

Un estándar integral que define las siete fases de un pentest profesional: acuerdo inicial, recopilación de inteligencia, análisis de amenazas, análisis de vulnerabilidades, explotación, post-explotación y reporte. PTES proporciona una estructura que asegura que ningún aspecto crítico sea omitido durante la evaluación.

OSSTMM (Open Source Security Testing Methodology Manual)

Un manual científico enfocado en la verificación y métrica de la seguridad operacional. OSSTMM se centra en proporcionar mediciones cuantitativas de la seguridad, permitiendo comparaciones objetivas entre diferentes sistemas y evaluaciones a lo largo del tiempo.

- **Concepto Clave:** La metodología asegura resultados consistentes, completos y profesionales. Sin un enfoque estructurado, es fácil pasar por alto vulnerabilidades críticas o realizar evaluaciones incompletas que no reflejen el verdadero estado de seguridad del sistema.

Fase 1: Network Discovery

Identificación de Activos en el Segmento de Red

El descubrimiento de red es el primer paso fundamental en cualquier evaluación de seguridad. Antes de poder atacar un sistema, debemos identificar qué dispositivos están presentes y accesibles en la red objetivo.

Objetivo Principal

Descubrir hosts activos en la red del laboratorio antes de realizar escaneos dirigidos. Este proceso nos permite:

- Mapear la topología básica de la red
- Identificar rangos de IP activos
- Reducir el tiempo de escaneos posteriores
- Evitar alertas innecesarias en sistemas no objetivos

Protocolo Utilizado

ARP (Address Resolution Protocol) para redes locales. ARP es especialmente efectivo en entornos de laboratorio porque:

- Opera en la capa 2 del modelo OSI
- No puede ser bloqueado por firewalls tradicionales
- Genera menos ruido en los logs del sistema
- Proporciona información sobre la tabla ARP local

Herramientas de Network Discovery

netdiscover

Herramienta versátil que puede operar tanto en modo pasivo como activo. En modo pasivo, escucha el tráfico ARP en la red sin generar paquetes, mientras que en modo activo sondea activamente enviando requests ARP a rangos específicos de IP.

- Modo pasivo: -p (stealth, sin generar tráfico)
- Modo activo: -r (especifica rango de red)
- Salida organizada por fabricante del NIC

arp-scan

Herramienta especializada que envía paquetes ARP a todos los hosts en la red local para descubrir dispositivos activos. Ofrece mayor control sobre el proceso de escaneo y opciones avanzadas de timing.

- Escaneo rápido y eficiente
- Detección de NICs duplicadas
- Base de datos actualizada de fabricantes

- **Ventaja Clave:** Ambas herramientas son más rápidas y sigilosas que un ping scan de nmap en redes locales, ya que operan a nivel de enlace de datos y no dependen de respuestas ICMP que pueden estar bloqueadas.

Práctica 1: Identificación del Objetivo

Ejecución del Descubrimiento de Red

1 Preparación del Entorno

Asegúrense de que tanto Kali Linux como Metasploitable 2 estén ejecutándose en la misma red virtual. Verifiquen la configuración de red de sus VMs para confirmar que pueden comunicarse entre sí.

2 Determinación del Rango de Red

Primero, identifiquen el rango de red de su entorno ejecutando `ip route` o `ifconfig` en su terminal de Kali. Anoten la dirección de red y la máscara de subred.

3 Ejecución del Comando

Instrucción: Desde su terminal de Kali, ejecuten:

```
sudo netdiscover -r  
<SU_RANGO_DE_RED>/24
```

Por ejemplo: `sudo netdiscover -r
192.168.1.0/24`

Objetivo: Los alumnos deben identificar y documentar la dirección IP de la VM Metasploitable 2. Busquen una entrada que corresponda a la MAC address de VirtualBox o VMware, dependiendo del hipervisor utilizado.

Fase 2: Port Scanning con nmap

Mapeo de Puertos, Servicios y Versiones

Una vez identificado el objetivo, el siguiente paso crítico es realizar un mapeo exhaustivo de los servicios disponibles. Esta fase determinará la superficie de ataque completa del sistema objetivo.

Objetivo Detallado

Identificar todos los puertos TCP/UDP abiertos, el software que se ejecuta en ellos y su versión exacta. Esta información es fundamental porque:

- Cada puerto abierto representa un punto de entrada potencial
- Las versiones específicas pueden tener vulnerabilidades conocidas
- La combinación de servicios puede crear vectores de ataque únicos
- Algunos servicios pueden estar mal configurados por defecto

Técnica Principal

TCP SYN Scan (-sS)

El escaneo SYN es el método por defecto para usuarios con privilegios de root. También conocido como "half-open scan", ofrece:

- Velocidad superior
- Menos detección en logs
- Menor impacto en servicios
- Compatible con la mayoría de sistemas

nmap: Parámetros Esenciales

Construyendo un Escaneo Efectivo



-p- (Escaneo Completo de Puertos)

Escanea los 65,535 puertos TCP disponibles. Por defecto, nmap solo escanea los 1,000 puertos más comunes, pero un atacante real probaría todos los puertos. Algunos servicios críticos pueden ejecutarse en puertos no estándar para "seguridad por oscuridad".



-sV (Detección de Versión)

Realiza una detección de versión del servicio conectándose a los puertos abiertos y analizando las respuestas del banner. Esta información es crucial para identificar vulnerabilidades específicas de versión en bases de datos como CVE.



-sC (Scripts por Defecto)

Ejecuta los scripts de enumeración por defecto del Nmap Scripting Engine (NSE). Estos scripts realizan checks básicos de seguridad, detección de vulnerabilidades comunes y recopilación de información adicional sin ser intrusivos.



-oN (Salida Normal)

Guarda la salida en un archivo de texto plano para su posterior análisis. Mantener registros detallados es esencial para documentar hallazgos, generar reportes profesionales y revisar resultados durante el proceso de explotación.

Práctica 2: Lanzamiento del Escaneo Principal

Ejecución del Escaneo de Servicios

Preparación del Comando

Ahora que tienen la IP de Metasploitable 2, van a lanzar un escaneo exhaustivo que nos proporcionará toda la información necesaria para las siguientes fases del pentesting.

Ejecución del Escaneo

Comando completo:

```
sudo nmap -p- -sV -sC -oN  
nmap_completo.txt <IP_MS2>
```

Sustituyan <IP_MS2> por la dirección IP real de su instancia de Metasploitable 2.

Instrucción: Mientras el escaneo se ejecuta, observen la información que va apareciendo en tiempo real. Nmap mostrará puertos abiertos conforme los detecte, junto con información preliminar sobre los servicios.

Tiempo de Ejecución

Este escaneo puede tardar varios minutos en completarse debido al escaneo completo de puertos. Utilicen este tiempo para repasar conceptos teóricos o preparar herramientas adicionales.

Análisis Inicial de Resultados

Primer Vistazo a la Superficie de Ataque

Una vez completado el escaneo, es momento de interpretar los resultados y identificar los vectores de ataque más prometedores. Metasploitable 2 está intencionalmente configurado con múltiples vulnerabilidades para fines educativos.

Revisión Colaborativa

En conjunto, examinaremos las primeras líneas del archivo `nmap_completo.txt` utilizando comandos como:

```
head -50 nmap_completo.txt  
grep "open" nmap_completo.txt
```

Buscaremos patrones y servicios que destaque como objetivos de alto valor.

Servicios de Alto Riesgo Típicos

- **FTP (Puerto 21)** - Posibles credenciales anonymous
- **Telnet (Puerto 23)** - Protocolo sin cifrado
- **SMTP (Puerto 25)** - Potencial relay abierto
- **HTTP (Puerto 80)** - Aplicaciones web vulnerables
- **SMB (Puerto 445)** - Vulnerabilidades de compartición
- **MySQL (Puerto 3306)** - Base de datos expuesta

Identificación

Catalogar todos los servicios detectados y sus versiones específicas

Investigación

Buscar vulnerabilidades conocidas para cada servicio identificado

1

2

3

4

Priorización

Clasificar servicios por nivel de riesgo y facilidad de explotación

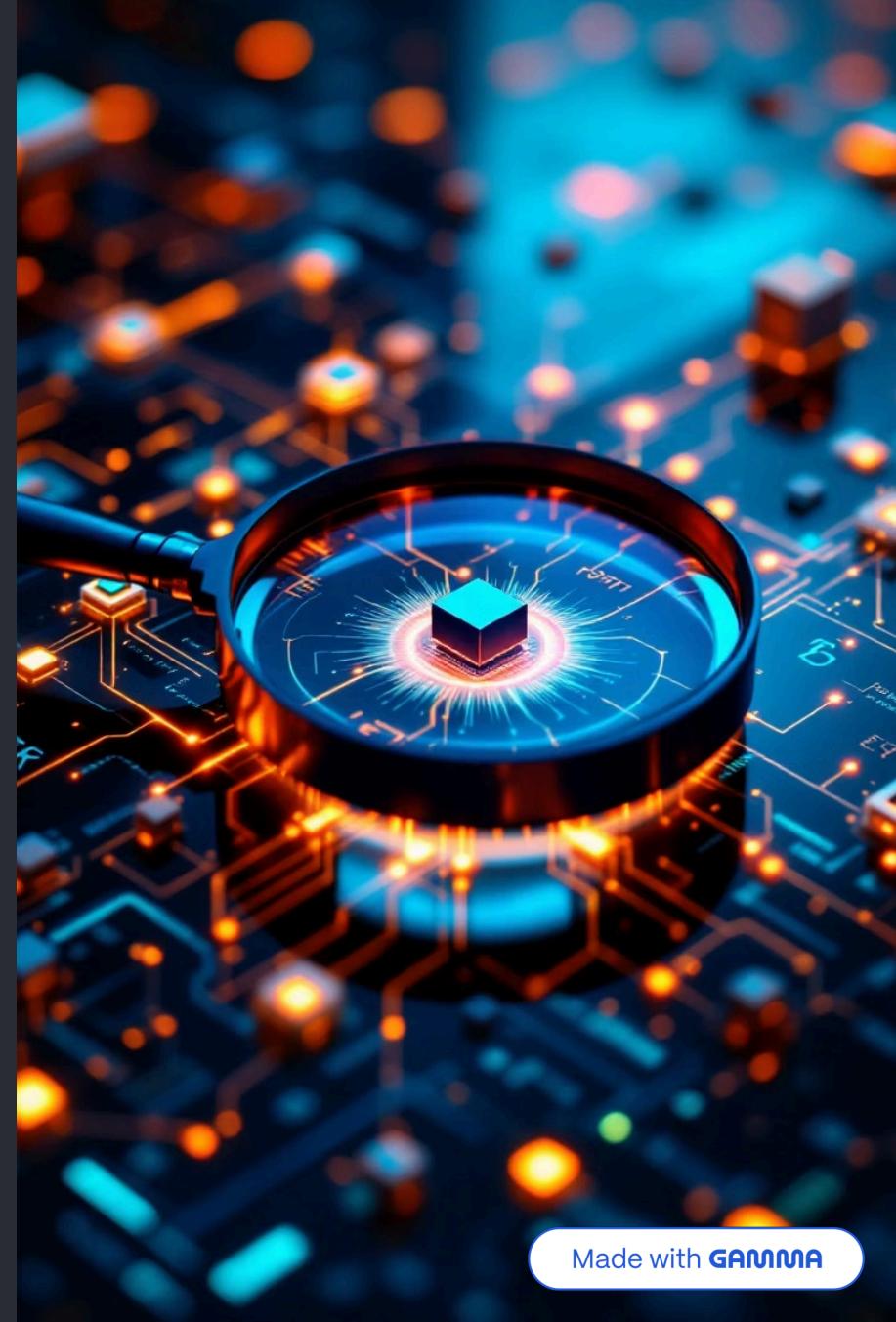
Planificación

Desarrollar estrategia de ataque basada en los hallazgos

- **Próximo Paso:** En el siguiente bloque profundizaremos en la enumeración específica de cada servicio identificado, utilizando herramientas especializadas para obtener información detallada que nos permita proceder con la fase de explotación.

Enumeración Sistemática de Servicios

Transformando el reconocimiento pasivo en inteligencia activa a través de la interacción directa con servicios identificados



De la Identificación a la Interacción

La enumeración sistemática representa el puente crítico entre el reconocimiento pasivo y la explotación activa. Mientras que el escaneo de puertos nos revela qué servicios están ejecutándose, la enumeración nos permite extraer información detallada que permanece oculta en un análisis superficial.

Este proceso implica la interacción activa y metodológica con cada servicio descubierto para obtener datos cruciales como configuraciones específicas, usuarios existentes, recursos compartidos accesibles, versiones exactas de software y posibles vectores de ataque. La información recopilada durante esta fase determina directamente el éxito de las etapas posteriores del pentesting.

La enumeración efectiva requiere un enfoque sistemático, documentación rigurosa y la comprensión profunda de cómo cada protocolo maneja la información sensible. Es aquí donde la experiencia del pentester se vuelve fundamental para distinguir entre datos irrelevantes y inteligencia actionable.

Enumeración de FTP: El Vector Anónimo

Análisis de vsftpd 2.3.4

El protocolo FTP, especialmente en versiones legacy como vsftpd 2.3.4, frecuentemente permite acceso anónimo como característica "conveniente" que los administradores olvidan deshabilitar.

Vector de Ataque: Acceso no autenticado a recursos del servidor

□ Comando práctico:

ftp <IP_MS2>

user: anonymous

pass: anonymous

Una vez conectados, debemos explorar sistemáticamente los directorios accesibles, documentar archivos sensibles y evaluar los permisos de escritura que podrían permitir la subida de payloads maliciosos.



Documentación del hallazgo: "Acceso anónimo a FTP permitido - Evaluar contenido y permisos de escritura"

Telnet: Credenciales por Defecto

Vulnerabilidad Crítica

Telnet transmite credenciales en texto plano y frecuentemente mantiene configuraciones por defecto en entornos de laboratorio como Metasploitable.

Vector de Explotación

Las credenciales msfadmin:msfadmin proporcionan acceso directo a shell de usuario, representando una vulnerabilidad de severidad crítica.

Implicaciones de Seguridad

El acceso vía Telnet con credenciales por defecto permite escalación de privilegios, movimiento lateral y persistencia en el sistema comprometido.

Comando de verificación: telnet <IP_MS2>

La combinación de protocolo inseguro y credenciales por defecto representa uno de los vectores de ataque más directos en pentesting. Este hallazgo debe documentarse como crítico y priorizarse en la fase de explotación.

SMTP: Enumeración de Usuarios

Postfix y la Fuga de Información

Los servidores SMTP legacy, particularmente versiones antiguas de Postfix, implementan comandos como VRFY y EXPN que permiten verificar la existencia de usuarios del sistema sin autenticación.

Esta funcionalidad, diseñada originalmente para verificar direcciones de correo válidas, se convierte en un vector de reconocimiento que expone información crítica sobre la estructura organizacional del objetivo.

"Los servidores de correo antiguos permitían verificar la existencia de usuarios. Esta funcionalidad representa una fuga de información significativa."

Metodología de enumeración:

1. Conectar: telnet <IP_MS2> 25
2. Verificar usuarios: VRFY root, VRFY msfadmin
3. Expandir listas: EXPN administrators
4. Documentar respuestas positivas



Hallazgo documentado: "El servicio SMTP permite la enumeración de usuarios del sistema"

Enumeración Web con Gobuster

1

Descubrimiento de Contenido Oculto

Gobuster utiliza ataques de fuerza bruta basados en diccionarios para descubrir directorios, archivos y recursos web que no están enlazados públicamente pero existen en el servidor.

2

Configuración Estratégica

La selección apropiada de wordlists y extensiones de archivo determina la efectividad del descubrimiento. Utilizamos dirb/common.txt con extensiones php y html para maximizar cobertura.

3

Análisis de Resultados

Los directorios descubiertos como /dvwa, /mutillidae, y /phpinfo.php revelan aplicaciones vulnerables instaladas intencionalmente para práctica de pentesting.

Comando ejecutado:

```
gobuster dir -u http://<IP_MS2> -w /usr/share/wordlists/dirb/common.txt -x php,html
```

Cada directorio descubierto representa una superficie de ataque potencial que debe ser investigada individualmente para identificar vulnerabilidades específicas y vectores de explotación.

Análisis Automatizado con Nikto

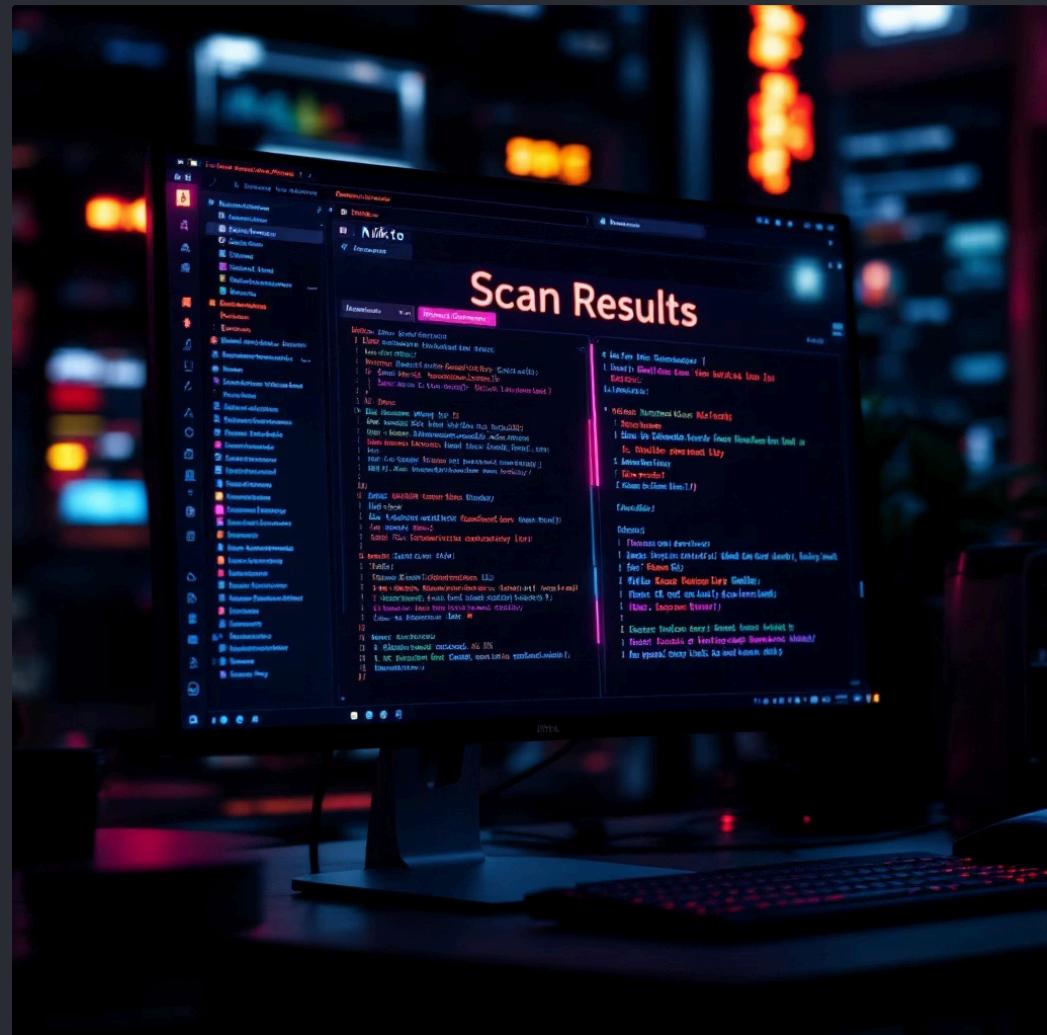
Escáner Especializado

Nikto es un escáner de vulnerabilidades web que analiza servidores HTTP/HTTPS identificando configuraciones peligrosas, archivos obsoletos, programas inseguros y versiones de software con vulnerabilidades conocidas.

Su base de datos contiene más de 6,700 elementos potencialmente peligrosos, incluyendo CVEs específicos, malas configuraciones comunes y archivos que no deberían ser accesibles públicamente.

Áreas de análisis clave:

- Versiones de servidor y módulos desactualizados
- Archivos de configuración expuestos
- Scripts administrativos accesibles
- Vulnerabilidades conocidas por CVE
- Headers de seguridad faltantes



Comando: `nikto -h http://<IP_MS2>`

Los resultados de Nikto deben correlacionarse con los hallazgos de Gobuster para crear un mapa completo de la superficie de ataque web del objetivo.



Enumeración Profunda de SMB

Samba 3.0.20: Vector Crítico

SMB es notoriamente complejo y frecuentemente mal configurado. Versiones legacy como Samba 3.0.20 contienen vulnerabilidades conocidas y configuraciones por defecto inseguras.

Los recursos compartidos anónimos pueden exponer información sensible del sistema, credenciales almacenadas y estructuras organizacionales completas.

enum4linux-ng: Herramienta Especializada

Esta herramienta realiza enumeración exhaustiva de sistemas Linux/Unix vía SMB, extrayendo información que incluye:

- Usuarios y grupos del sistema
- Recursos compartidos y permisos
- Políticas de contraseñas
- Información del sistema operativo
- Servicios NetBIOS disponibles

Análisis de Resultados

La salida extensa de enum4linux-ng debe analizarse sistemáticamente, priorizando shares accesibles anónimamente y usuarios con nombres predictibles.

Comando: enum4linux-ng -A <IP_MS2>

Documentación Sistemática: El Cerebro del Pentest



Estructura Organizacional

Organizad cada hallazgo por servicio y puerto, manteniendo una estructura consistente que permita referencia rápida durante las fases posteriores del pentest.



Priorización de Vectores

Clasificad cada hallazgo según su severidad y facilidad de explotación. Las credenciales por defecto y accesos anónimos son vectores de alta prioridad.



Timestamps y Contexto

Registrad el momento exacto de cada descubrimiento y las condiciones específicas que permitieron el hallazgo para futura referencia y replicación.

La documentación no es simplemente un registro de actividades; es el recurso más valioso del pentester. Cada comando ejecutado, cada respuesta obtenida y cada conclusión derivada debe capturarse de manera que permita la construcción de un mapa mental completo del objetivo.

"A medida que ejecutáis cada comando, copiad los resultados relevantes en vuestro archivo de notas. Este documento es vuestro recurso más valioso para las fases de explotación y post-explotación."

Consolidación: De Datos a Inteligencia



Servicios Enumerados

FTP, Telnet, SMTP, HTTP y SMB completamente analizados



Vectores Críticos

Acceso anónimo FTP, credenciales por defecto
Telnet, enumeración SMTP



Superficie Mapeada

Inventario completo de servicios y vulnerabilidades potenciales

Checklist de Verificación Completado

01

Interacción Manual Exitosa

FTP, Telnet y SMTP enumerados manualmente con hallazgos significativos documentados

02

Enumeración Web Automatizada

Gobuster y Nikto ejecutados exitosamente, directorios ocultos y vulnerabilidades identificadas

03

SMB Completamente Analizado

enum4linux-ng proporcionó inteligencia detallada sobre usuarios y recursos compartidos

04

Documentación Sistemática

Todos los hallazgos organizados y priorizados para la fase de explotación

La enumeración sistemática ha transformado un conjunto de puertos abiertos en un mapa detallado de vectores de ataque. Cada servicio analizado ha revelado configuraciones inseguras, credenciales por defecto o exposición de información que facilitará la explotación exitosa en las fases posteriores del pentesting.

Correlación de Vulnerabilidades y Formulación de Vectores de Ataque

En esta fase crítica del pentesting, transformamos los hallazgos de reconocimiento en planes de ataque estructurados y priorizados. Exploraremos cómo conectar versiones de software identificadas con exploits conocidos, utilizando herramientas especializadas para construir una estrategia de penetración efectiva.

Conectando Hallazgos con Exploits Conocidos

Definición del Proceso

La correlación de vulnerabilidades es el proceso metodológico de tomar la información obtenida durante la fase de enumeración —principalmente versiones específicas de software y servicios— y cruzarla sistemáticamente con bases de datos públicas de vulnerabilidades para identificar exploits funcionales y verificados.

Importancia Estratégica

Este paso marca la transición del reconocimiento pasivo a la preparación activa para el compromiso. No se trata simplemente de encontrar vulnerabilidades, sino de identificar aquellas que son explotables y que proporcionan el mayor impacto con el menor riesgo de detección.

Flujo de Trabajo

1. **Recopilación de versiones** identificadas durante la enumeración
2. **Consulta sistemática** en bases de datos de exploits
3. **Verificación de funcionalidad** y compatibilidad de exploits
4. **Documentación detallada** de CVEs y módulos disponibles
5. **Priorización** basada en impacto y probabilidad de éxito



searchsploit: Tu Arsenal Local



Base de Datos Offline

searchsploit mantiene una copia local completa de la base de datos Exploit Database, permitiendo búsquedas rápidas sin conexión a Internet y evitando la exposición de nuestras investigaciones.



Interfaz de Línea de Comandos

Herramienta eficiente que permite filtrado avanzado por plataforma, tipo de exploit, fecha de publicación y palabras clave específicas del software objetivo.



Análisis de Código

Capacidad de examinar el código fuente completo de los exploits directamente, permitiendo comprensión profunda del vector de ataque antes de la ejecución.

Comando Esencial para Mantenimiento

```
searchsploit --update
```

Este comando asegura que nuestra base de datos local esté sincronizada con las últimas vulnerabilidades publicadas. La actualización regular es crucial para mantener la efectividad de nuestro arsenal de exploits y no perder oportunidades de compromiso en sistemas recién vulnerados.

Caso de Estudio 1: vsftpd 2.3.4

Investigación Inicial

```
searchsploit vsftpd 2.3.4
```

Análisis del Exploit

La investigación revela una vulnerabilidad crítica conocida como "smiley face backdoor". Esta puerta trasera fue introducida maliciosamente en el código fuente de vsftpd entre las fechas 20110630 y 20110703.

Mecanismo de Explotación

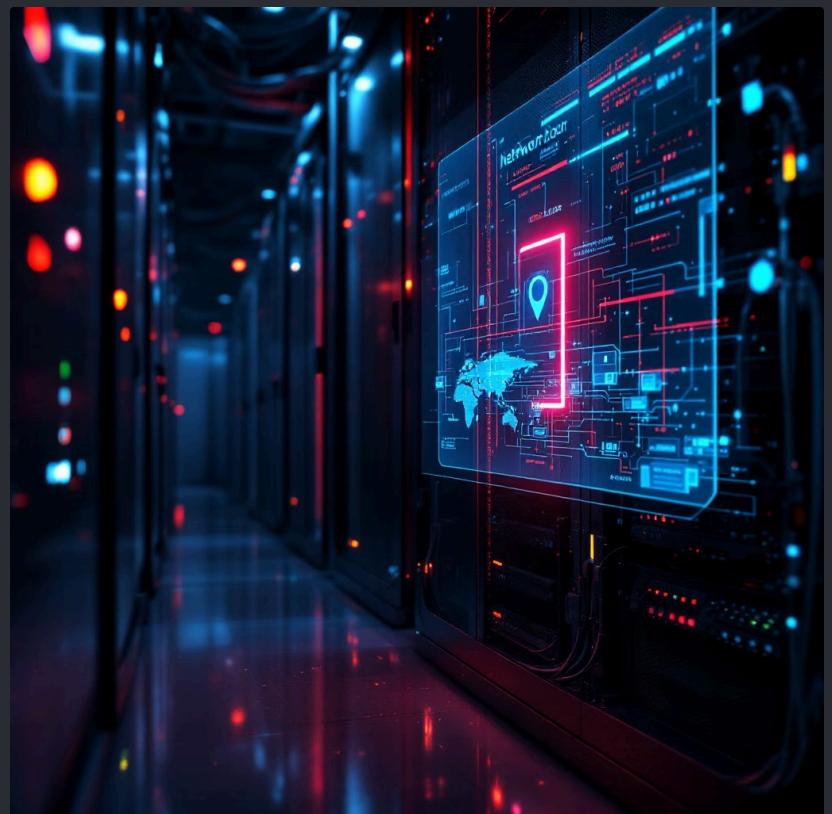
El exploit se activa cuando un usuario envía un nombre de usuario que contiene la secuencia ":" seguido de cualquier carácter. Esta acción desencadena la apertura de un shell en el puerto 6200 con privilegios de root.

Documentación Técnica

- **CVE:** CVE-2011-2523
- **Módulo Metasploit:** exploit/unix/ftp/vsftpd_234_backdoor
- **Puerto de Backdoor:** 6200/TCP
- **Privilegios Obtenidos:** root

Comando para examinar el código:

```
searchsploit -x /usr/share/exploitdb/exploits/unix/remote/17491.rb
```



Impacto Crítico

Esta vulnerabilidad representa uno de los compromisos más directos posibles: acceso root inmediato sin autenticación previa. La facilidad de explotación y el alto privilegio obtenido la convierten en vector primario.



Caso de Estudio 2: Samba 3.0.20

1

Investigación

```
searchsploit samba 3.0.20
```

La búsqueda identifica múltiples exploits, destacando el "Username map script" como vector principal de compromiso para esta versión específica.

2

Análisis Técnico

El exploit aprovecha una vulnerabilidad en el manejo del parámetro "username map script" en smb.conf. Permite la ejecución de comandos shell mediante metacaracteres en el nombre de usuario durante la autenticación SMB.

3

Proof of Concept

El PoC demuestra cómo injectar comandos utilizando la sintaxis "/bin/sh" seguida del comando deseado. El payload se ejecuta con los privilegios del demonio smbd, típicamente root en configuraciones por defecto.

4

Documentación Final

- **CVE:** CVE-2007-2447
- **Módulo Metasploit:** exploit/multi/samba/usermap_script
- **Puertos Afectados:** 139/445 TCP
- **Tipo:** Remote Command Execution

Caso de Estudio 3: UnrealIRCd

Investigación de la Vulnerabilidad

```
searchsploit unrealircd
```

UnrealIRCd, un popular servidor de IRC, fue comprometido con una puerta trasera similar al caso de vsftpd. Esta backdoor afecta específicamente a las versiones 3.2.8.1 distribuidas entre noviembre 2009 y junio 2010.

Mecanismo de Activación

La backdoor se activa cuando un cliente envía el comando "AB" seguido de comandos del sistema. El servidor ejecuta estos comandos con los privilegios del usuario que ejecuta el demonio IRC, frecuentemente root en sistemas mal configurados.

Vector de Explotación

A diferencia de otros exploits que requieren autenticación o protocolos específicos, esta vulnerabilidad puede ser explotada simplemente conectándose al puerto IRC (típicamente 6667) y enviando la secuencia maliciosa.

Documentación Técnica

- **Versiones Afectadas:** 3.2.8.1 (distribuciones comprometidas)
- **Puerto por Defecto:** 6667/TCP
- **Módulo MSF:** exploit/unix/irc/unreal ircd_3281_backdoor
- **Privilegios:** Usuario del demonio (potencialmente root)
- **Complejidad:** Muy baja - No requiere autenticación

▢ Contexto Histórico

Esta backdoor fue descubierta cuando un mirror oficial fue comprometido, distribuyendo versiones maliciosas durante meses. Ilustra la importancia de verificar integridad de software descargado.

Del Hallazgo a la Estrategia



Identificación

Hallazgo de vulnerabilidades específicas durante la fase de enumeración y correlación con bases de datos públicas.

Análisis

Evaluación detallada del impacto, complejidad de explotación y probabilidad de éxito de cada vulnerabilidad identificada.

Formulación

Construcción de un plan detallado y paso a paso que maximice las posibilidades de compromiso exitoso del sistema objetivo.

Definición de Vector de Ataque

Un Vector de Ataque es un plan metodológico y detallado para comprometer un sistema, basado en una vulnerabilidad o debilidad específica identificada. Incluye la secuencia exacta de acciones, herramientas requeridas, payloads a utilizar, y resultados esperados en cada etapa del proceso de compromiso.

La formulación efectiva de vectores de ataque requiere considerar factores como la detección por sistemas de seguridad, la persistencia post-compromiso, las rutas de escalación de privilegios, y las opciones de movimiento lateral dentro de la red objetivo.

Vector de Ataque N°1: Credenciales Débiles



Servicio Objetivo

Telnet (Puerto 23/TCP)

Protocolo de acceso remoto sin cifrado que permite conexión directa al shell del sistema.



Debilidad Identificada

Credenciales por Defecto

Sistema configurado con credenciales predecibles: msfadmin:msfadmin



Resultado Esperado

Shell de Usuario

Acceso interactivo inmediato con privilegios de usuario estándar para reconocimiento interno.

Plan de Acción Detallado

- Conexión inicial:** Establecer sesión telnet al puerto 23
- Autenticación:** Proporcionar credenciales msfadmin:msfadmin
- Verificación de acceso:** Confirmar shell activo y funcional
- Reconocimiento local:** Identificar usuarios, grupos y permisos
- Enumeración del sistema:** Mapear estructura de directorios y servicios locales

Comando de Ejecución

```
telnet 192.168.1.101 23
```



Prioridad CRÍTICA

Este vector obtiene la máxima prioridad debido a su simplicidad de ejecución, alta probabilidad de éxito (100% si las credenciales no han sido cambiadas), y bajo riesgo de detección por sistemas de monitoreo.

Vector de Ataque N°2: Explotación de Servicio SMB

01

Preparación del Exploit

Cargar y configurar el módulo exploit/multi/samba/usermap_script en Metasploit Framework, estableciendo el target y payload apropiados para el sistema objetivo.

03

Ejecución del Exploit

Enviar payload malicioso aprovechando CVE-2007-2447 en el parámetro username durante autenticación SMB para ejecutar comandos con privilegios elevados.

02

Configuración de Payload

Seleccionar payload/cmd/unix/reverse que establecerá conexión de vuelta a nuestro sistema de ataque, evitando restricciones de firewall saliente.

04

Establecimiento de Shell

Recibir conexión reverse shell con privilegios de root, proporcionando control administrativo completo sobre el sistema comprometido.

Especificaciones Técnicas

- **Servicio:** Samba SMB (Puertos 139/445 TCP)
- **Vulnerabilidad:** CVE-2007-2447 ("Username map script")
- **Herramienta:** Metasploit Framework
- **Módulo:** exploit/multi/samba/usermap_script
- **Payload Recomendado:** cmd/unix/reverse
- **Privilegios Obtenidos:** root (UID 0)
- **Prioridad:** ALTA

Consideraciones de Ejecución

Este vector requiere que el servicio Samba esté configurado con "username map script" habilitado. La explotación es confiable pero genera logs más visibles que el vector de credenciales débiles.

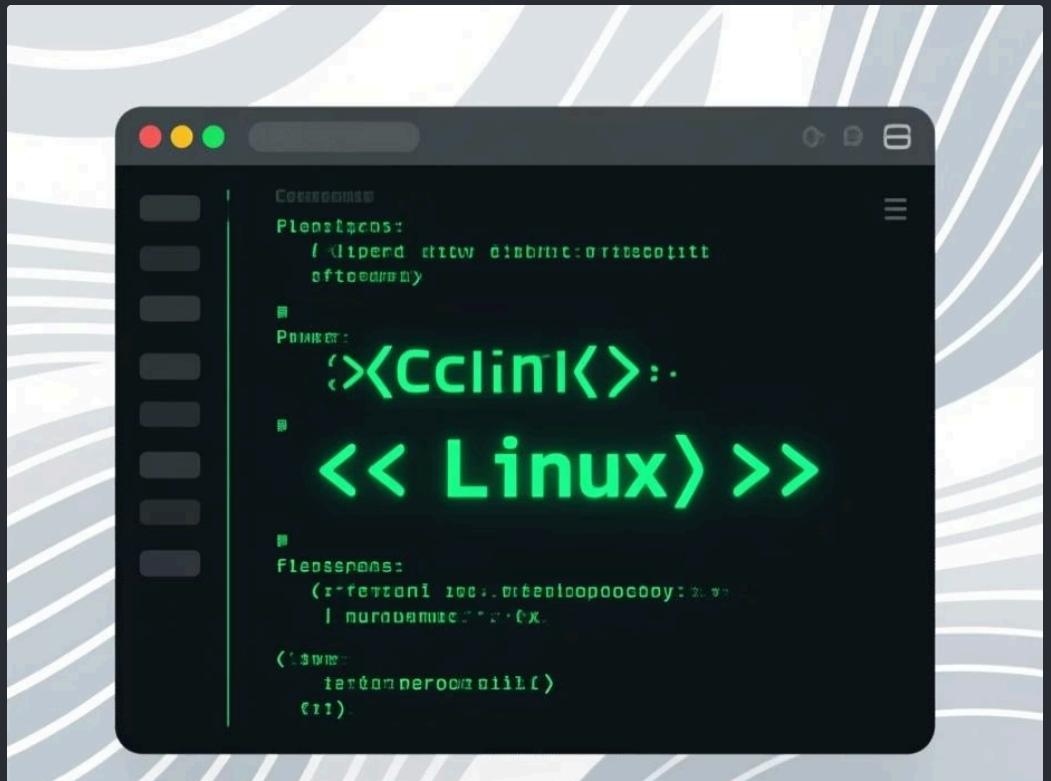
El acceso root inmediato elimina la necesidad de escalación de privilegios, permitiendo acceso completo a configuraciones, archivos sensibles, y capacidad de instalación de persistencia o backdoors adicionales.

El Plan de Batalla está Trazado

Trabajo del Analista Completado

Hemos transformado exitosamente una simple dirección IP en un informe técnico comprehensivo con múltiples vectores de ataque documentados, priorizados y listos para ejecución. Este proceso metodológico representa el núcleo del trabajo profesional de un pentester.

Metodología Aplicada



El Próximo Capítulo

En las próximas semanas, en la primera clase del **Taller de Ciberseguridad Ofensiva**, nuestra misión será ejecutar estos vectores de ataque metodológicamente, uno por uno, y obtener control total del sistema objetivo.

Pasaremos de la teoría a la práctica, transformando planes en compromiso real del sistema Metasploitable 2.

Espacio para Preguntas

¿Dudas sobre la metodología aplicada? ¿Consultas técnicas sobre algún vector específico? ¿Preguntas sobre herramientas utilizadas?