ATAQUES DE FUERZA BRUTA

SI NO SE PUEDE A LAS BUENAS... A LAS MALAS!

1 Presentación del curso

Qué vamos a ver

- En este curso sobre Hydra y ataques de fuerza bruta, veremosa fondo una de las técnicas esenciales en auditorías de seguridad y pruebas de penetración.

- Hydra es una herramienta poderosa que permite realizar ataques de fuerza bruta en una amplia gama de servicios, desde SSH y FTP hasta HTTP y SMB. Aprenderemos cómo utilizar Hydra de manera efectiva y ética para descubrir vulnerabilidades de autenticación y fortalecer la seguridad en sistemas y redes.

- El curso está diseñado para ofrecer tanto fundamentos teóricos como aplicaciones prácticas de Hydra en distintos escenarios. Abordaremos desde ataques básicos de fuerza bruta hasta técnicas avanzadas, como el uso de diccionarios personalizados y el ajuste de parámetros de rendimiento.

WHOAMI

- +5 años de experiencia en el mundo del Hacking y Ciberseguridad
- Divulgador en Redes
- Proyectos nacionales e internacionales
- Experiencia y factores diferenciadores

Requisitos previos

- Conocimiento en Redes
- Conocimientos en Linux
- Conocimientos Básicos de Hacking
- Conocimientos de NMAP

Laboratorio

Configuración del entorno de laboratorio en Virtual Box

- Kali Linux / Ubuntu
- Metasploitable 2

2
Introducción a Hydra y Fuerza Bruta

Ataques de Fuerza Bruta

Un ataque de fuerza bruta es un método que implica intentar todas las combinaciones posibles de contraseñas hasta encontrar la correcta.

Este enfoque se basa en la premisa de que, dado el tiempo suficiente, cualquier contraseña puede ser descubierta probando sistemáticamente todas las combinaciones posibles.

Fuerza bruta VS ataques de diccionario

Mientras que los ataques de fuerza bruta prueban todas las combinaciones posibles de caracteres, los ataques de diccionario utilizan una lista predefinida de contraseñas comunes o probables.

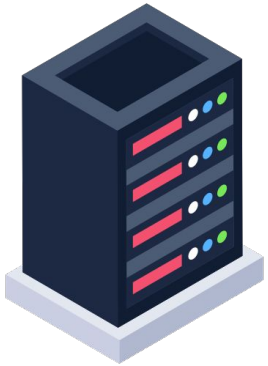
Los ataques de diccionario son más rápidos pero menos exhaustivos, ya que se limitan a las contraseñas incluidas en el diccionario utilizado.

Password Spraying

Password Spraying es una técnica de ataque en la que se utiliza una contraseña común en múltiples cuentas de usuario en lugar de probar muchas contraseñas en una sola cuenta.

Este tipo de ataque reduce el riesgo de detección, ya que se evita bloquear cuentas por múltiples intentos fallidos en un solo usuario.

Es particularmente efectivo cuando los atacantes tienen acceso a listas de usuarios, como en redes corporativas.



Importancia en pentest y hacking

Los ataques de fuerza bruta son cruciales para evaluar la fortaleza de las contraseñas y la seguridad de los sistemas.

Permiten identificar vulnerabilidades que podrían ser explotadas por atacantes, proporcionando información valiosa para mejorar las políticas de contraseñas y la configuración de seguridad.

PERO MUCHO CUIDADO!!

Es importante advertir que los ataques de fuerza bruta, como los realizados con Hydra, pueden tener efectos adversos en los sistemas objetivo.

Estos ataques generan una gran cantidad de intentos de inicio de sesión en un corto período, lo que puede causar estrés en el servidor y llevar a su desestabilización o a la degradación del rendimiento.

Además, muchos sistemas están configurados para bloquear cuentas tras varios intentos fallidos de inicio de sesión, lo que podría resultar en el bloqueo temporal o permanente de usuarios legítimos.

Por estas razones, es esencial realizar estos ataques únicamente en entornos controlados y con autorización explícita, asegurándose de aplicar las técnicas de manera responsable y con un enfoque en la seguridad y la integridad del sistema.

¿Qué es Hydra?

Hydra es una herramienta de código abierto diseñada para realizar ataques de fuerza bruta en diversos protocolos y servicios, facilitando la obtención de contraseñas.

Es ampliamente utilizada en pruebas de penetración y auditorías de seguridad para evaluar la robustez de las credenciales de acceso.

Historia y desarrollo de Hydra

Hydra fue desarrollada por el equipo de The Hacker's Choice (THC) como una solución para auditar la seguridad de contraseñas en aplicaciones y servicios.

A lo largo de los años, ha evolucionado para soportar múltiples protocolos y configuraciones, convirtiéndose en una herramienta esencial en el arsenal de los profesionales de la seguridad informática.

Instalación en Linux

Instalación en Kali Linux y Parrot OS (preinstalado):

- Hydra viene preinstalada en estas distribuciones orientadas a la seguridad, facilitando su uso inmediato para pruebas de seguridad. Los usuarios pueden verificar su presencia ejecutando el comando hydra -h en la terminal.

Instalación en Ubuntu y otras distribuciones de Linux:

- Se puede instalar utilizando el gestor de paquetes APT con el comando sudo apt install hydra. Es recomendable actualizar los repositorios antes de la instalación ejecutando sudo apt update.
    - sudo apt install hydra

Comandos para verificar la correcta instalación de Hydra:

- Ejecuta hydra -v en la terminal para ver la versión instalada y confirmar que la instalación fue exitosa. Si Hydra está correctamente instalado, este comando mostrará la versión actual y las opciones disponibles.

3

Uso Básico de Hydra

Sintaxis y Opciones Principales

La estructura general es:

hydra [opciones] [servicio]://[objetivo].

Por ejemplo, para atacar un servicio SSH en una dirección IP específica, el comando sería:

hydra [opciones] ssh://[dirección_ip]

Descripción de las opciones más utilizadas:

- **-l <usuario>**: Especifica un único nombre de usuario para el ataque.
- **-L <archivo_usuarios>**: Indica un archivo que contiene una lista de nombres de usuario.
- **-p <contraseña>**: Define una única contraseña a probar.
- **-P <archivo_contraseñas>**: Señala un archivo con una lista de contraseñas.

Sintaxis y Opciones Principales

- **-M <archivo_objetivos>**: Indica un archivo que contiene una lista de objetivos (direcciones IP o dominios).
-
- **-t /T <número>:** Determina el número de tareas (hilos) en paralelo; el valor predeterminado es 16.
- **-v / -V:** Activa el modo detallado; -v muestra información básica y -V proporciona detalles de cada intento.
- **-f:** Finaliza el ataque al encontrar la primera combinación válida de usuario y contraseña.
- **-o <archivo_salida>:** Guarda los resultados en el archivo especificado.
  - **-b:** especificar el formato
- **-s <puerto>:** Especifica un puerto diferente al predeterminado para el servicio objetivo.
- **-e nsr:** Prueba contraseñas vacías (n), iguales al nombre de usuario (s) o el nombre de usuario invertido (r).

Sintaxis y Opciones Principales

- **-x <min>:<max>:<charset>:** Genera contraseñas en el rango de longitud especificado (min a max) utilizando el conjunto de caracteres definido (charset).
- **-R:** Restaura una sesión anterior desde el archivo hydra.restore.
- **-S:** Fuerza el uso de SSL/TLS para conexiones.
  - **-O:** usa SSLv2 y SSLv3
- **-I:** Ignora errores de conexión de sesiones anteriores y continúa el ataque.
- **-W <número>:** Espera el número de segundos especificado entre cada intento.
- **-K:** Mantiene las conexiones abiertas; útil para protocolos que lo permiten.
- **-F:** Finaliza el ataque al encontrar una combinación válida en cualquier objetivo (cuando se usa con -M).
- **-u:** Alterna el orden de prueba; en lugar de probar todas las contraseñas para un usuario antes de pasar al siguiente, prueba una contraseña para todos los usuarios y luego pasa a la siguiente contraseña.

Sintaxis y Opciones Principales

- **-w <tiempo>:** Establece el tiempo máximo de espera por respuesta en segundos; el valor predeterminado es 30 segundos.
- **-q:** Silencia la salida estándar, mostrando solo resultados válidos.
- **-d:** Activa el modo de depuración, proporcionando información detallada para diagnóstico.
- **-U:** Muestra información detallada sobre los módulos y protocolos soportados.
- **-h:** Muestra la ayuda y todas las opciones disponibles.
- **-C <archivo_combo>:** usar un archivo con el combo usuario:contraseña

Realización de un Ataque de Fuerza Bruta Simple

Un comando BÁSICO podría ser:

- **hydra -l usuario -P lista_contrasenas.txt ssh://direccion_ip.**

Este comando intentará iniciar sesión en el servicio SSH de la dirección IP especificada utilizando el nombre de usuario proporcionado y probando cada contraseña de la lista.

Análisis de los resultados obtenidos:

Los resultados mostrarán qué contraseñas fueron probadas y cuáles fueron efectivas, permitiendo evaluar la fortaleza de las contraseñas utilizadas.

4

Uso Avanzado de Hydra

Creación de usuarios vulnerables

Creación de usuario

- **useradd -m -s /bin/bash usuario_nuevo:** creación de usuario
- **passwd:** para establecer la contraseña.

Usuarios vulnerables

- **manolo:**<BLANK>
- **kike:**ekik
- **msfadmin**:msfadmin
- **user**:user
- **postgres**:postgres
- **sys**:batman
- **klog**:123456789
- **service**:service

Ataques de Fuerza Bruta contra Múltiples Protocolos

- **FTP**: hydra -l usuario -P lista_contrasenas.txt ftp://direccion_ip.
- **SSH:** hydra -l usuario -P lista_contrasenas.txt ssh://192.168.1.100
- **HTTP**: hydra -l usuario -P lista_contrasenas.txt http://direccion_ip.
- **POP3**: hydra -l usuario -P lista_contrasenas.txt pop3://192.168.1.100 -s 110
- **SMTP**: hydra -l usuario -P lista_contrasenas.txt smtp://192.168.1.100 -s 25
- **IMAP**: hydra -l usuario -P lista_contrasenas.txt imap://192.168.1.100 -s 143
- **MYSQL**: hydra -l usuario -P lista_contrasenas.txt mysql://192.168.1.100
- **RDP**: hydra -l usuario -P lista_contrasenas.txt rdp://192.168.1.100
- **POSTGRESQL**: hydra -P lista_contrasenas.txt postgres://192.168.1.100
- **SMB**: hydra -l usuario -P lista_contrasenas.txt smb://direccion_ip.

4.1

Casos de uso

Contraseñas Vacías y Mismas que el Nombre de Usuario

hydra -L usuarios.txt -e nsr ftp://192.168.1.40 -V -t 4 -o resultados_vacios.txt

Ataca SSH probando contraseñas vacías y contraseñas iguales al nombre de usuario.

Opciones usadas:

- **-L usuarios.txt:** Lista de usuarios.
- **-e ns:** Prueba contraseñas vacías (n), contraseñas iguales al nombre de usuario (s) y nombre de usuario al revés (r).
- **-V:** Modo detallado.
- **-t 4:** Utiliza 4 tareas en paralelo.
- **-o resultados_vacios.txt:** Guarda los resultados en resultados_vacios.txt.

Ataque Generando Contraseñas Personalizadas

hydra -l usuario -x 6:8:abc123 ftp://192.168.1.30 -V -t 5 -o resultados_personalizados.txt

Ataca FTP generando contraseñas personalizadas.

Opciones usadas:

- **-l usuario:** Nombre de usuario específico (usuario).
- **-x 6:8:abc123:** Genera contraseñas entre 6 y 8 caracteres usando el conjunto abc123.
- **-V:** Modo detallado.
- **-t 5:** Utiliza 5 tareas en paralelo.
- **-o resultados_personalizados.txt:** Guarda los resultados en resultados_personalizados.txt.

Múltiples Objetivos con Diferentes Usuarios y Contraseñas

hydra -L usuarios.txt -P contraseñas.txt -M objetivos.txt ftp -s 21 -f -V -o resultados.txt

Realiza un ataque de fuerza bruta en el servicio FTP en varios objetivos especificados en objetivos.txt, utilizando listas de usuarios y contraseñas.

Opciones usadas:

- **-L usuarios.txt:** Lista de usuarios.
- **-P contraseñas.txt:** Lista de contraseñas.
- **-M objetivos.txt:** Lista de objetivos (direcciones IP o dominios).
- **-s 21:** Utiliza el puerto 21, específico de FTP.
- **-f:** Finaliza el ataque en cada objetivo al encontrar una combinación válida.
- **-V:** Modo detallado.
- **-o resultados.txt:** Guarda los resultados en resultados.txt.

Ataque con Alternancia en el Orden de Pruebas

hydra -L usuarios.txt -P contraseñas.txt postgres://192.168.1.10 -u -t 6 -o resultados_postgres.txt

Ataca un servicio SSH en la IP 192.168.1.10 utilizando listas de usuarios y contraseñas, alternando el orden de las pruebas.

Opciones usadas:

- **-L usuarios.txt:** Lista de usuarios.
- **-P contraseñas.txt:** Lista de contraseñas.
- **-u:** Alterna el orden de pruebas (prueba cada contraseña con todos los usuarios antes de pasar al siguiente).
- **-t 6:** Utiliza 6 tareas en paralelo.
- **-o resultados_ssh.txt:** Guarda los resultados en resultados_ssh.txt.

Ataque con Combo List

hydra -C defaults.txt ftp://192.168.1.10 -t 6 -o resultados_ftp.txt

Ataca un servicio SSH en la IP 192.168.1.10 utilizando listas de usuarios y contraseñas, alternando el orden de las pruebas.

Opciones usadas:

- **-C defaults.txt:** Lista de combos.
- **-t 6:** Utiliza 6 tareas en paralelo.
- **-o resultados_ftp.txt:** Guarda los resultados en resultados_ftp.txt.

Ataques a Formularios Web

Sintaxis básica:

- **hydra -l usuario -P lista_contrasenas.txt http://direccion_ip.**

Es importante determinar cómo se envían los datos (POST/GET) y qué campos se utilizan.

Ataques a Formularios Web

Ejemplo de comando para un ataque GET:

- **hydra -l admin -P contraseñas.txt http-get-form "/login?usuario=^USER^&contrasena=^PASS^:F=Acceso denegado"**

Donde:

- **http-get-form:** Especifica que el ataque será mediante el método GET.
- **/ruta_login?usuario=^USER^&contrasena=^PASS^:** La URL del formulario de inicio de sesión, con parámetros de usuario y contraseña.
- **F=mensaje_error**: Define el mensaje de error que aparece cuando falla la autenticación.

Ataques a Formularios Web

Ejemplo de comando para un ataque POST:

- **hydra -l usuario -P lista_contrasenas.txt direccion_ip http-post-form "/ruta_login:campo_usuario=^USER^&campo_contrasena=^PASS^:mensaje_error"**
- **hydra -l admin -P /ruta/a/lista_contrasenas.txt 192.168.1.100 http-post-form "/login.php:username=^USER^&password=^PASS^:F=Error de autenticación"**

Donde:

- **direccion_ip:** La dirección IP o dominio del objetivo.
- **http-post-form:** Define que se utilizará el método POST.
- **/ruta_login:** La ruta del formulario de inicio de sesión.
- **campo_usuario=^USER^&campo_contrasena=^PASS^:** Los campos del formulario que corresponden al usuario y la contraseña.
- **mensaje_error:** El texto que aparece cuando la autenticación falla.

4.2

Creación de diccionarios

Crunch - Uso de Diccionarios Personalizados

Crunch es una herramienta muy versátil para generar diccionarios basados en reglas específicas, como longitud de palabras, caracteres permitidos, etc.

Ejemplo de comando con Crunch:

crunch 8 12 abcdef123 -o diccionario.txt

- **8 12:** Especifica la longitud mínima (8) y máxima (12) de las palabras generadas.
- **abcdef123:** Caracteres que se incluirán en las combinaciones.
- **-o diccionario.txt:** Guarda el diccionario en un archivo llamado diccionario.txt.

Cewl - Uso de Diccionarios Personalizados

Cewl es una herramienta que permite crear diccionarios personalizados a partir de contenido de una página web. Es muy útil para generar diccionarios relacionados con términos específicos de un sitio objetivo.

Ejemplo de comando con Cewl:

cewl -d 2 -m 5 https://example.com -w diccionario.txt

- **-d 2:** recursividad de búsqueda
- **-m 5:** mínimo de longitud de las contraseñas
- **https://example.com:** La URL del sitio objetivo.
- **-w diccionario.txt:** Guarda las palabras extraídas en un archivo llamado diccionario.txt.

Diccionarios Online/Offline

Si no necesitas generar un diccionario desde cero, puedes usar repositorios de listas de contraseñas como **Weakpass** y **SecLists** que contienen diccionarios preconstruidos, incluyendo listas de palabras comunes, combinaciones de patrones, y listas filtradas para distintos contextos.

- **ONLINE:**
    - **SecLists GitHub:** https://github.com/danielmiessler/SecLists/tree/master/Passwords
    - **Weakpass:** https://weakpass.com/

O también puedes usar los predefinidos en Kali Linux:

- **OFFLINE:** generalmente bajo /usr/share/wordlist
    - **Rockyou**: offline, sobretodo para CTFs

5

Optimización y Tiempos

Ajuste de Parámetros para Mejorar la Eficiencia

Configuración del número de tareas paralelas (-t):

Ajustar el número de hilos puede acelerar el ataque; el valor predeterminado es 16, pero puede aumentarse según la capacidad del sistema.

Manejo de tiempos de espera y reintentos:

Utiliza opciones como -w para establecer tiempos de espera y -W espera X segundo entre conexiones para cada hilo.

Ataque Interrumpido y Añadir Tiempo de Espera entre Intentos

hydra -R -w 10 -L usuarios.txt -P contraseñas.txt ssh://192.168.1.40 -t 4 -o resultados_restaurados.txt

Restaura un ataque SSH interrumpido, con un tiempo de espera de 10 segundos entre intentos.

Opciones usadas:

- **-R:** Restaura una sesión anterior desde el archivo hydra.restore.
- **-w 10:** Tiempo máximo de espera de 10 segundos por respuesta.
- **-L usuarios.txt:** Lista de usuarios.
- **-P contraseñas.txt:** Lista de contraseñas.
- **-t 4:** Utiliza 4 tareas en paralelo.
- **-o resultados_restaurados.txt:** Guarda los resultados en resultados_restaurados.txt.

6

Otras Herramientas

MEDUSA

Medusa: Introducción

Medusa es una herramienta de fuerza bruta modular diseñada para realizar pruebas en múltiples servicios y protocolos de red.

- **sudo apt-get install medusa**

Medusa: Opciones Principales (Parte 1)

- **-h:** Muestra la ayuda y lista todas las opciones disponibles.
- **-H <archivo_objetivos>:** Define un archivo que contiene una lista de objetivos (IPs o nombres de host).
- **-h <IP o dominio>:** Especifica un solo objetivo (dirección IP o nombre de host).
- **-U <archivo_usuarios>**: Define un archivo con una lista de nombres de usuario a probar.
- **-u <usuario>:** Especifica un único nombre de usuario.
- **-P <archivo_contraseñas>:** Indica un archivo que contiene una lista de contraseñas a probar.
- **-p <contraseña>:** Define una única contraseña.
- **-M <módulo>:** Especifica el módulo de servicio que se quiere atacar, como SSH, FTP, HTTP, etc.

Medusa: Opciones Principales (Parte 2)

- **-n <puerto>:** Selecciona un puerto específico para el servicio (si es diferente al predeterminado).
- **-t <número>:** Establece el número de tareas en paralelo; el valor por defecto es 16.
- **-T <tiempo>:** Define el tiempo de espera para cada conexión (en segundos).
- **-f:** Detiene el ataque al encontrar la primera combinación válida de usuario y contraseña.
- **-r <reintentos>:** Número de intentos de reconexión en caso de fallo.
- **-e <n/p>:** Prueba contraseñas vacías (n) o contraseñas iguales al nombre de usuario (p).
- **-o <archivo_salida>:** Guarda los resultados en el archivo especificado.

Ejemplo de Ataque SSH con Medusa

Ataque al servicio SSH en el puerto 22 contra el usuario admin usando una lista de contraseñas.

- **medusa -h 192.168.1.10 -u admin -P /path/to/passwords.txt -M ssh -n 22**

Ejemplo de Ataque FTP con Medusa

Ataque al servicio FTP en el puerto 21, usando listas de usuarios y contraseñas.

- **medusa -h 192.168.1.20 -U /path/to/users.txt -P /path/to/passwords.txt -M ftp -n 21**

Ejemplo de Ataque HTTP con Medusa

Ataque a un formulario web HTTP en el puerto 80 con el usuario testuser y una lista de contraseñas, ajustando el número de tareas concurrentes a 5.

- **medusa -h targetwebsite.com -u testuser -P /path/to/passwords.txt -M http -n 80 -T 5**

Ejemplo de Ataque SMB con Medusa

Ataque al servicio SMB usando listas de usuarios y contraseñas para intentar acceder a una carpeta de red.

- **medusa -h 192.168.1.30 -U /path/to/users.txt -P /path/to/passwords.txt -M smbnt**

NCRACK

Ncrack: Introducción

Ncrack es una herramienta de fuerza bruta enfocada en la auditoría de autenticación de red, desarrollada por el equipo de Nmap.

- **https://www.kali.org/tools/ncrack/**

Ncrack: Opciones Principales (Parte 1)

- **-h:** Muestra ayuda y lista de opciones.
- **-p <protocolo>:<puerto>:** Especifica el protocolo y el puerto a atacar. Ejemplo: p ssh:22.
- **-U <archivo_usuarios>:** Define un archivo con una lista de nombres de usuario.
- **-P <archivo_contraseñas>:** Indica un archivo que contiene una lista de contraseñas a probar.
- **-user <usuario>:** Define un único usuario.
- **-pass <contraseña>:** Define una única contraseña.
- **-iL <archivo_objetivos>:** Carga una lista de objetivos desde un archivo.
- **-g <tiempo_espera>:** Ajusta el tiempo de espera para cada intento (en segundos).

Ncrack: Opciones Principales (Parte 2)

- **-T <número_tareas>:** Número de tareas concurrentes por protocolo.
- **--min-parallelism <número>:** Número mínimo de tareas concurrentes.
- **--max-parallelism <número>:** Número máximo de tareas concurrentes.
- **--delay <milisegundos>:** Añade un retraso entre intentos para cada tarea.

Ncrack: Opciones de Salida

- **-oN <archivo_salida>:** Guarda los resultados en un archivo en formato normal.
- **-oX <archivo_salida>:** Guarda los resultados en formato XML.
- **-v / vv:** Incrementa la verbosidad; útil para obtener detalles sobre cada intento.

Ejemplo de Ataque a RDP con Ncrack

Ataque a un servidor RDP en la IP 192.168.1.15 con el usuario Administrator y una lista de contraseñas.

- **ncrack -p rdp://192.168.1.15 -u Administrator -P /path/to/passwords.txt**

Ataque SSH con Usuarios y Contraseñas Múltiples

Ataque SSH en el puerto 22 a múltiples objetivos especificados en el archivo targets.txt con listas de usuarios y contraseñas.

- **ncrack -p ssh:22 -iL /path/to/targets.txt -U /path/to/users.txt -P /path/to/passwords.txt**

Ejemplo de Ataque HTTP con Paralelismo Ajustado

Ataque HTTP en el puerto 80 con el usuario testuser, usando una lista de contraseñas y ajustando el paralelismo entre 2 y 5 tareas.

- **ncrack -p http:80 -u testuser -P /path/to/passwords.txt --min-parallelism 2 --max-parallelism 5**

Ejemplo de Ataque a Múltiples Servicios Simultáneos

Ataque simultáneo a SSH y FTP en la IP 192.168.1.10 usando el usuario root y una lista de contraseñas.

- **ncrack -p ssh://192.168.1.10,ftp://192.168.1.10 -u root -P /path/to/passwords.txt**

BRUTEX

NETEXEC

Netexec: Introducción

Netexec es una herramienta de administración de redes que permite ejecutar comandos de manera remota en múltiples sistemas simultáneamente, facilitando la gestión y el control de redes grandes.

También puede utilizarse en auditorías de seguridad para probar la autenticación de servicios de red.

Netexec: Opciones Principales

- **IP o rango:** Especifica un archivo con una lista de hosts para atacar.
- **-u <usuario>:** Define el nombre de usuario para el ataque o un listado.
- **-p <contraseñas.txt>:** Específica contraseña o archivo un archivo que contiene una lista de contraseñas.
- **--continue-on-succes:** Continúa el ataque incluso después de encontrar credenciales válidas.
- **--no-brute:** Desactiva el modo de fuerza bruta tradicional, ya que estamos probando una correspondencia directa entre usuarios y contraseñas.

Ataque de Fuerza Bruta con Netexec en SMB

Realiza un ataque de fuerza bruta a hosts con servicio SMB, probando la autenticación con el usuario 'admin' y una lista de contraseñas. Si la autenticación es exitosa, ejecuta el comando 'dir' para listar archivos en el directorio remoto.

- **netexec IP -u admin -P contraseñas.txt -o resultados_smb.txt**

Consideraciones de Seguridad

Tanto el ataque de fuerza bruta como el Password Spraying son técnicas potentes en auditorías de seguridad, pero deben emplearse éticamente y con autorización.

Es fundamental monitorear los resultados y realizar estos ataques de manera controlada para evitar bloqueos de cuenta y minimizar el riesgo de detección en entornos de prueba.

METASPLOIT

Introducción a la Fuerza Bruta en Metasploit

Metasploit es una plataforma avanzada de pruebas de penetración que incluye módulos para realizar ataques de fuerza bruta en varios servicios.

Estos módulos permiten probar combinaciones de usuarios y contraseñas en servicios críticos como FTP, SSH y SMB, ayudando a identificar credenciales débiles en entornos controlados.

Metasploit: Módulo de Fuerza Bruta para FTP

El módulo **auxiliary/scanner/ftp/ftp_login** permite realizar ataques de fuerza bruta en servidores FTP probando combinaciones de usuarios y contraseñas.

Este módulo es útil para identificar credenciales débiles en servicios FTP, que suelen estar configurados en redes internas o para acceder a archivos compartidos.

Comando para iniciar el módulo: **use auxiliary/scanner/ftp/ftp_login**

Opciones clave:

- **RHOSTS**: Especifica la IP o el rango de IPs objetivo.
- **USER_FILE**: Define un archivo con una lista de nombres de usuario.
- **PASS_FILE**: Especifica un archivo con una lista de contraseñas.
- **STOP_ON_SUCCESS**: Detiene el ataque al encontrar una combinación válida.

Ejemplo de Ataque de Fuerza Bruta en FTP

Realiza un ataque de fuerza bruta en el servicio FTP en la IP 192.168.1.10, utilizando listas de usuarios y contraseñas.

- use auxiliary/scanner/ftp/ftp_login
- set RHOSTS 192.168.1.10
- set USER_FILE /path/to/users.txt
- set PASS_FILE /path/to/passwords.txt
- run

Metasploit: Módulo de Fuerza Bruta para SSH

El módulo **auxiliary/scanner/ssh/ssh_login** permite realizar ataques de fuerza bruta en servidores SSH.

SSH es un protocolo comúnmente utilizado para administrar sistemas de manera remota, y este módulo ayuda a identificar credenciales débiles en este servicio.

Comando para iniciar el módulo: **use auxiliary/scanner/ssh/ssh_login**

Opciones clave:

- **RHOSTS**: Especifica la IP o el rango de IPs objetivo.
- **USER_FILE**: Define un archivo con una lista de nombres de usuario.
- **PASS_FILE**: Especifica un archivo con una lista de contraseñas.
- **STOP_ON_SUCCESS**: Detiene el ataque al encontrar una combinación válida.

Ejemplo de Ataque de Fuerza Bruta en SSH

Ataca el servicio SSH en la IP 192.168.1.20 probando diferentes combinaciones de usuario y contraseña para acceder.

- use auxiliary/scanner/ssh/ssh_login
- set RHOSTS 192.168.1.20
- set USER_FILE /path/to/users.txt
- set PASS_FILE /path/to/passwords.txt
- run

Metasploit: Módulo de Fuerza Bruta para SMB (smb_login)

El módulo **auxiliary/scanner/smb/smb_login** permite realizar ataques de fuerza bruta en servicios SMB.

SMB es un protocolo utilizado para compartir archivos y recursos en redes Windows, y este módulo ayuda a probar accesos no autorizados en estos sistemas.

Comando para iniciar el módulo: **use auxiliary/scanner/smb/smb_login**

Opciones clave:

- **RHOSTS**: Especifica la IP o el rango de IPs objetivo.
- **USER_FILE**: Define un archivo con una lista de nombres de usuario.
- **PASS_FILE**: Especifica un archivo con una lista de contraseñas.
- **STOP_ON_SUCCESS**: Detiene el ataque al encontrar una combinación válida.
- **SMB::Domain:** Especifica el dominio SMB si es necesario.

Ejemplo de Ataque de Fuerza Bruta en SMB

Realiza un ataque de fuerza bruta en el servicio SMB en la IP 192.168.1.30 para acceder a los recursos compartidos utilizando diferentes combinaciones de usuarios y contraseñas.

- use auxiliary/scanner/smb/smb_login
- set RHOSTS 192.168.1.30
- set USER_FILE /path/to/users.txt
- set PASS_FILE /path/to/passwords.txt
- run

NMAP NSE BRUTEFORCE

Introducción a la Fuerza Bruta con Nmap y NSE

Nmap es una herramienta de escaneo de redes muy potente que, gracias a su motor de scripts (NSE - Nmap Scripting Engine), permite realizar ataques de fuerza bruta en diversos servicios.

El uso de scripts NSE enfocados en fuerza bruta permite que Nmap ataque servicios como FTP, SSH y SMB para identificar credenciales débiles, facilitando auditorías de seguridad en entornos controlados.

NSE para Fuerza Bruta en FTP

El script **ftp-brute** realiza ataques de fuerza bruta contra servicios FTP.

Permite probar combinaciones de usuarios y contraseñas en servidores FTP.

Comando básico:

- **nmap -p 21 --script ftp-brute <IP>**

Es posible usar archivos personalizados de usuarios y contraseñas.

- **nmap -p 21 --script ftp-brute --script-args userdb=/root/userdb.txt,passdb=/root/passdb.txt,ftp-brute.threads=5 <IP>**

Opciones clave:

- **ftp-brute.threads:** Define el número de hilos para realizar el ataque (por defecto es 10).
- **userdb:** Especifica un archivo de texto con una lista de nombres de usuario.
- **passdb:** Indica un archivo con una lista de contraseñas a probar.

NSE para Fuerza Bruta en otros Protocolos

Proporciona un método para probar múltiples credenciales en los distintos protocolos.

- nmap -p 22 --script **ssh-brute** <IP>
- nmap -sU -p 161 --script **snmp-brute** <IP>
- nmap -p 3306 --script **mysql-brute** <IP>
- nmap -p 445 --script **smb-brute** --script-args userdb=/root/userdb.txt,passdb=/root/passdb.txt <IP>

7

Próximos Pasos

Y ahora... Qué?

- Explorar a fondo otras herramientas
  - Curso de Metasploit
  - Curso de NMAP
- Preparación de certificación sencilla como el eJPT