Aquí está el desglose del comando:

ffuf: Es el nombre de la herramienta.

-u http://192.168.0.7:1336/FFUZ: Aquí se especifica la URL objetivo. La parte FFUZ es un marcador de posición que ffuf reemplazará con las entradas de la lista de palabras (wordlist).

-w /usr/share/seclists/Discovery/web-Content/directory-list-lowercase-2.3-small.txt: Aquí se especifica la wordlist (lista de palabras) que se utilizará para la fuerza bruta. Esta lista contiene nombres de archivos y directorios comunes en minúsculas que ffuf probará contra la URL objetivo.

¿Qué hace este comando?

ffuf enviará peticiones HTTP al servidor en http://192.168.0.7:1336/, reemplazando FFUZ en la URL con cada una de las entradas en la wordlist especificada.

Por ejemplo, si la wordlist contiene las palabras "admin", "login", y "config", ffuf intentará acceder a http://192.168.0.7:1336/admin, http://192.168.0.7:1336/login, y http://192.168.0.7:1336/config.

El objetivo es descubrir qué archivos o directorios existen en el servidor web.

Resultado esperado

La herramienta mostrará qué URLs dieron una respuesta válida del servidor, lo que indica que esos archivos o directorios existen.

Esta es una técnica común en pruebas de penetración para descubrir recursos ocultos o no documentados en un servidor web.

**Comandos y Opciones Comunes de ffuf**

1. **-u URL (URL):**  
   Especifica la URL objetivo donde FUZZ será reemplazado por las entradas de la lista de palabras.

bash

Copiar código

ffuf -u http://example.com/FUZZ

1. **-w WORDLIST (Wordlist):**  
   Especifica la lista de palabras a utilizar para fuzzing.

bash

Copiar código

ffuf -w /path/to/wordlist.txt

1. **-r (Follow redirects):**  
   Sigue las redirecciones HTTP automáticamente.

bash

Copiar código

ffuf -r -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-recursion (Recursion):**  
   Realiza fuzzing recursivo en los directorios encontrados.

bash

Copiar código

ffuf -recursion -u http://example.com/FUZZ/ -w /path/to/wordlist.txt

1. **-e EXTENSIONS (Extensions):**  
   Especifica las extensiones a agregar a las rutas durante el fuzzing. Separa múltiples extensiones con comas.

bash

Copiar código

ffuf -e .php,.html,.txt -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-o OUTPUT\_FILE (Output file):**  
   Especifica un archivo para guardar los resultados del fuzzing.

bash

Copiar código

ffuf -o results.txt -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-mc CODES (Match HTTP status codes):**  
   Filtra los resultados para mostrar solo aquellos que coincidan con los códigos de estado HTTP especificados.

bash

Copiar código

ffuf -mc 200,301 -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-fc CODES (Filter HTTP status codes):**  
   Excluye los resultados que coincidan con los códigos de estado HTTP especificados.

bash

Copiar código

ffuf -fc 404 -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-fs SIZE (Filter by response size):**  
   Filtra los resultados excluyendo las respuestas que coincidan con el tamaño de respuesta especificado en bytes.

bash

Copiar código

ffuf -fs 1024 -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-fr REGEX (Filter by regex):**  
   Excluye los resultados que coincidan con la expresión regular especificada en el cuerpo de la respuesta.

bash

Copiar código

ffuf -fr "Access Denied" -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-t THREADS (Threads):**  
   Establece el número de hilos (threads) para realizar el fuzzing. Más hilos aumentan la velocidad, pero también la carga sobre el servidor objetivo.

bash

Copiar código

ffuf -t 40 -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-timeout SECONDS (Timeout):**  
   Establece el tiempo máximo de espera para cada solicitud en segundos.

bash

Copiar código

ffuf -timeout 10 -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-b COOKIE (Cookies):**  
   Especifica cookies para la solicitud HTTP. Útil para fuzzing en sesiones autenticadas.

bash

Copiar código

ffuf -b "SESSIONID=abcd1234" -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-H HEADER (HTTP headers):**  
   Añade cabeceras HTTP personalizadas a las solicitudes. Puede utilizarse varias veces para múltiples cabeceras.

bash

Copiar código

ffuf -H "User-Agent: Mozilla/5.0" -H "Referer: http://example.com" -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-X METHOD (HTTP method):**  
   Especifica el método HTTP a utilizar (GET, POST, etc.).

bash

Copiar código

ffuf -X POST -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-d DATA (POST data):**  
   Utiliza esta opción junto con el método POST para enviar datos en el cuerpo de la solicitud.

bash

Copiar código

ffuf -X POST -d "username=admin&password=FUZZ" -u http://example.com/login -w /path/to/wordlist.txt

1. **-p DELAY (Request delay):**  
   Establece un retraso en segundos entre cada solicitud, útil para evitar ser bloqueado por el servidor.

bash

Copiar código

ffuf -p 0.5 -u http://example.com/FUZZ -w /path/to/wordlist.txt

1. **-v (Verbose):**  
   Muestra más detalles durante la ejecución.

bash

Copiar código

ffuf -v -u http://example.com/FUZZ -w /path/to/wordlist.txt

**Ejemplo Completo**

bash

Copiar código

ffuf -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u http://example.com/FUZZ -e .php,.html -r -t 50 -o results.txt -mc 200

Este comando realiza lo siguiente:

* Usa una lista de palabras para directorios.
* Fuzzea en la URL especificada.
* Prueba con extensiones .php y .html.
* Sigue las redirecciones.
* Utiliza 50 hilos para acelerar el proceso.
* Guarda los resultados en results.txt.
* Solo muestra los resultados con código HTTP 200.

Estas son las opciones más comunes y útiles de ffuf que puedes utilizar en tus pruebas de seguridad en Kali Linux.