

DirBiertete con DirB

URL BRUTEFORCER

a T.O.S. production by...
The DarkRaver

Introducción

- Presentación
- Descripción de la herramienta
- Ejemplos de uso
- Líneas futuras
- Conclusiones

¿Qué es DirB?

- DirB es un scanner web.
- Su objetivo principal es buscar objetos en un website.
- Funciona realizando un ataque de diccionario contra el webserver.

¿Para que sirve DirB?

- Básicamente para buscar objetos en un sitio web. 😊
- Especialmente para su posterior auditoria en busca de vulnerabilidades.
- También puede ser usado como scanner de CGI clásico, mediante el uso de una wordlist adecuada.

¿Diferencias con otros scanners?

- DirB no es una herramienta novedosa, pero hace cosas que otras no hacen:
 - ◆ DirB no busca vulnerabilidades, solo objetos para analizar manualmente.
 - ◆ DirB no trabaja como un web-spider. No sigue links como la mayoría de scanners. Trabaja solo mediante un ataque de diccionario.
 - ◆ DirB no descarga las webs ni analiza el código HTML (Bueno a veces si).

Objetivos del desarrollo

- Enfoque profesional
- Eficiencia
- Versatilidad
- Robustez

Enfoque profesional

- Nunca me han gustado las herramientas que te lo dan “todo” hecho:
 - ◆ Tienden a dar falsos positivos/negativos.
 - ◆ No funcionan en entornos no habituales.
 - ◆ No permiten al técnico explotar todas sus posibilidades.
- DirB esta pensado para ayudar en el trabajo de auditoria web, pero el trabajo final debe realizarlo un técnico cualificado.

Eficiencia

- DirB esta programado para intentar hacer bien su trabajo, no para hacer muchos trabajos.
- No es una “suite”, debe ser usada con el apoyo de otras herramientas.
- Su trabajo complementa el trabajo de otras herramientas sin duplicar tareas.

Versatilidad

- Otros scanners avanzados permiten ataques de diccionario. Pero:
 - ◆ No permiten elegir las extensiones a buscar.
 - ◆ Su recursividad es limitada.
 - ◆ No permiten añadirle una Cookie, una cabecera o alguna característica adicional a la petición.

Robustez

- DirB es capaz de trabajar casi con cualquier tipo de webserver.
- La mayoría de scanners no pueden trabajar en entornos no estándar.
- Se basan solamente en el código HTTP devuelto (Si es 404 o no).
- DirB hace un fingerprint del código devuelto cuando una pagina realmente no existe (NEC).

NEC

- NEC = Not Existant Code
- Una de las ventajas de DirB es que antes de comenzar un escaneo realiza un fingerprintg de las paginas que devuelve el servidor cuando un recurso es no encontrado.
- De esta forma en servidores que no devuelven un 404 Not Found, es posible diferenciar los resultados positivos de los negativos.

Historia

- DirB se concibió inicialmente como un sencillo scanner de directorios web (DirB = Directory Bruteforcer).
- Posteriormente se vio su utilidad y eficacia y se amplió para buscar también ficheros.
- Ahora ya permite el ataque de diccionario también contra URLs directamente (URL Bruteforcer).

Libwww o Libcurl

- En las primeras fases de desarrollo de DirB tuve que elegir que API utilizar.
- Un API desarrollado por mi mismo, tenia una serie de pegas:
 - ◆ Perdida de tiempo en desarrollar funciones que otros ya habían hecho.
 - ◆ Perdida de portabilidad.
 - ◆ Posibilidad de introducir mas bugs.
- Después de analizar las APIs HTTP disponibles la decisión se redujo a 2.

Libwww o Libcurl

- Finalmente me decidí por Libcurl por estas razones:
 - ◆ API más sencillo de programar.
 - ◆ Más funciones para tareas concretas.
 - ◆ Más rápido/eficiente/robusto.
 - ◆ ¿Más portable?
- Pegas:
 - ◆ No cuenta con API de parseo HTML.

¿Qué tiene DirB que lo hace una herramienta útil en el mundo real?

- El ataque es recursivo.
- Las wordlists están muy depuradas:
 - ◆ Son lo suficientemente pequeñas para no hacer muy lento el escaneo.
 - ◆ Son lo suficientemente completas para contener los nombres de fichero/directorio más comunes.
- Permite trabajar sobre casi cualquier formato de URL:
 - ◆ Sobre HTTPS. (P.Ej. <https://www.test.com/>)
 - ◆ Sobre directorios directamente. (P.Ej. <http://www.test.com/directorio/xxxx>)
 - ◆ Incluso sobre URLs extrañas. (P.Ej. <http://www.test.com/servlet?file=/xxxx>)

Ejemplos de uso:

- Escaneo normal:

```
$ ./dirb.exe http://www.ruboskizo.net/  
wordlists/common.txt,wordlists/spanish.txt -o  
../demo2.txt -X ,,.php
```

```
-----  
DIRB v1.3.1  
By The Dark Raver  
-----
```

```
OUTPUT_FILE: ../demo2.txt  
START_TIME: Fri Oct 8 16:31:37 2004  
URL_BASE: http://www.ruboskizo.net/  
WORDLIST_FILES: wordlists/common.txt,wordlists/spanish.txt  
EXTENSIONS_LIST: (,,.php) | ().(php)  
SERVER_BANNER: Apache/1.3.31 (Unix) PHP/4.3.8  
NOT_EXISTANT_CODE: 404
```

```
-----  
Generating Wordlist...  
Generated Words: 1707
```

```
---- Scanning URL: http://www.ruboskizo.net/ ----  
(* ) DIRECTORY: http://www.ruboskizo.net/demo/  
(* ) DIRECTORY: http://www.ruboskizo.net/images/  
FOUND: http://www.ruboskizo.net/index.php - CODE: 200  
(* ) DIRECTORY: http://www.ruboskizo.net/intranet/  
(* ) DIRECTORY: http://www.ruboskizo.net/src/  
(* ) DIRECTORY: http://www.ruboskizo.net/webmail/
```

```
---- Entering directory: http://www.ruboskizo.net/demo/ ----  
(* ) DIRECTORY: http://www.ruboskizo.net/demo/admon/  
FOUND: http://www.ruboskizo.net/demo/index.php - CODE: 200  
(* ) DIRECTORY: http://www.ruboskizo.net/demo/js/  
(* ) DIRECTORY: http://www.ruboskizo.net/demo/upload/  
(* ) DIRECTORY: http://www.ruboskizo.net/demo/user/
```

```
---- Entering directory: http://www.ruboskizo.net/images/ ----  
(!) WARNING: Directory is listable. No need to scan it.  
(Use mode -w if you want to scan it anyway)
```

```
---- Entering directory: http://www.ruboskizo.net/intranet/ ----  
(* ) DIRECTORY: http://www.ruboskizo.net/intranet/admin/  
FOUND: http://www.ruboskizo.net/intranet/index.php - CODE: 401  
(* ) DIRECTORY: http://www.ruboskizo.net/intranet/js/  
(* ) DIRECTORY: http://www.ruboskizo.net/intranet/mysql/  
(* ) DIRECTORY: http://www.ruboskizo.net/intranet/user/
```

Ejemplos de uso:

- Escaneo de un directorio:

```
$ ./dirb.exe http://www.ejemplo.net/servlet/  
wordlists/spanish.txt
```

Ejemplos de uso:

- Escaneo de una URL:

- ◆ Guía de televisión de ya.com:

- <http://television.ya.com/STv?M=main>

- ◆ Busquemos otras posibles opciones:

- ```
$./dirb.exe http://television.ya.com/STv?M=
wordlists/common.txt
```

-----  
DIRB v1.3.1  
By The Dark Raver  
-----

OUTPUT\_FILE: ../demo3.txt  
START\_TIME: Fri Oct 8 17:12:36 2004  
URL\_BASE: http://television.ya.com/STv?M=  
WORDLIST\_FILES: wordlists/common.txt  
OPTION: NOT forcing an ending '/' on URLs  
SERVER\_BANNER: Apache  
NOT\_EXISTANT\_CODE: 302

-----  
Generating Wordlist...  
Generated Words: 633

---- Scanning URL: http://television.ya.com/STv?M= ----  
FOUND: http://television.ya.com/STv?M=channel - CODE: 200  
FOUND: http://television.ya.com/STv?M=main - CODE: 200  
FOUND: http://television.ya.com/STv?M=program - CODE: 200  
FOUND: http://television.ya.com/STv?M=search - CODE: 200  
FOUND: http://television.ya.com/STv?M=test - CODE: 200

-----  
DOWNLOADED: 633 - FOUND: 5

# DirB puede salvarte el culo

- Desde que tengo DirB los marrones me parecen menos. 😊
- Son las 9 de la mañana, a mediodía tengo que presentar un informe sobre una web que solo tiene 4 HTMLs hechos con Frontpage.
- Los scanners de vulnerabilidades no sacan nada.
- ¿Qué hacer? Tu solución es DirB!.

```

DIRB v1.3.0
By The Dark Raver

```

```
START_TIME: Mon May 24 22:24:13 2004
URL_BASE: http://www.ejemplo.com/
WORDLIST_FILES: wordlists/common.txt,wordlists/spanish.txt
OPTION: Printing LOCATION header
EXTENSIONS_LIST: (.html,.jsp,,) | (.html)(.jsp)(
NOT_EXISTANT_CODE: 404
SERVER_BANNER: Microsoft-IIS/5.0

```

```
Generating Wordlist...
Generated Words: 2484
```

```
---- Scanning URL: http://www.ejemplo.com/ ----
(*) DIRECTORY: http://www.ejemplo.com/Administration/
FOUND: http://www.ejemplo.com/index.jsp - CODE: 500
FOUND: http://www.ejemplo.com/services - CODE: 200
```

```
---- Entering directory: http://www.ejemplo.com/Administration/ ----
(*) DIRECTORY: http://www.ejemplo.com/Administration/error/
(*) DIRECTORY: http://www.ejemplo.com/Administration/images/
(*) DIRECTORY: http://www.ejemplo.com/Administration/ssi/
FOUND: http://www.ejemplo.com/Administration/welcome.jsp - CODE: 200
```

```
---- Entering directory: http://www.ejemplo.com/Administration/error/ ----
```

```
---- Entering directory: http://www.ejemplo.com/Administration/images/ ----
```

```
---- Entering directory: http://www.ejemplo.com/Administration/ssi/ ----
FOUND: http://www.ejemplo.com/Administration/ssi/games.jsp - CODE: 200
FOUND: http://www.ejemplo.com/Administration/ssi/games - CODE: 200
FOUND: http://www.ejemplo.com/Administration/ssi/gest.html - CODE: 200
FOUND: http://www.ejemplo.com/Administration/ssi/gest.jsp - CODE: 200
FOUND: http://www.ejemplo.com/Administration/ssi/gest - CODE: 200
```

```

DOWNLOADED: 14904 - FOUND: 8
```

# Funcionalidades pendientes

- DirB tienes 2 líneas de desarrollo futuro principales:
  - ◆ Análisis de URLs.
    - ★ Hoy por hoy, DirB ya puede ser utilizado para ataques sencillos contra cualquier URL.
    - ★ Es necesario profundizar en ataques mas específicos. (SQL Injection, URI encoding injection, etc...)
  - ◆ Análisis de Application Servers.
    - ★ El modo Application Server permitiría analizar una web para distinguir el origen de los contenidos.

# Conclusiones

- DirB cubre un hueco que dejan otros scanners web.
- Este hueco es cada vez más importante:
  - ◆ Los aplicativos hechos a medida son cada vez más habituales.
  - ◆ Los contenidos web son cada vez más difíciles de catalogar para los scanners clásicos.
  - ◆ El hacking web es cada vez más popular.

# Conclusiones

- DirB es aun una herramienta joven, le queda mucho por mejorar y crecer.
- Aun así es ya una herramienta muy útil y permite obtener resultados muy satisfactorios.

# Esto es el fin...

- Dudas, consultas, ideas...
- ¿Quién se anima a ser beta-tester? 😊
- mailto: [darkraver@t0s.org](mailto:darkraver@t0s.org)

The Dark Raver