DIRB

February 18, 2014 ports Web Applications

DIRB Package Description

DIRB is a Web Content Scanner. It looks for existing (and/or hidden) Web Objects. It basically works by launching a dictionary based attack against a web server and analyzing the response.

DIRB comes with a set of preconfigured attack wordlists for easy usage but you can use your custom wordlists. Also DIRB sometimes can be used as a classic CGI scanner, but remember is a content scanner not a vulnerability scanner.

DIRB main purpose is to help in professional web application auditing. Specially in security related testing. It covers some holes not covered by classic web vulnerability scanners. DIRB looks for specific web objects that other generic CGI scanners can't look for. It doesn't search vulnerabilities nor does it look for web contents that can be vulnerables.

Source: http://dirb.sourceforge.net/about.html
DIRB Homepage | Kali DIRB Repo

- Author: The Dark Raver
- License: GPLv2

tools included in the dirb package

dirb – A web content scanner

```
root@kali:~# dirb


-----------------
DIRB v2.21
By The Dark Raver
-----------------

./dirb <url_base> [<wordlist_file(s)>] [options]

========================= NOTES =========================
 <url_base> : Base URL to scan. (Use -resume for session resuming)
 <wordlist_file(s)> : List of wordfiles. (wordfile1,wordfile2,wordfile3...)

======================== HOTKEYS ========================
 'n' -> Go to next directory.
 'q' -> Stop scan. (Saving state for resume)
 'r' -> Remaining scan stats.

======================== OPTIONS ========================
 -a <agent_string> : Specify your custom USER_AGENT.
 -c <cookie_string> : Set a cookie for the HTTP request.
 -f : Fine tunning of NOT_FOUND (404) detection.
 -H <header_string> : Add a custom header to the HTTP request.
 -i : Use case-insensitive search.
 -l : Print "Location" header when found.
 -N <nf_code>: Ignore responses with this HTTP code.
 -o <output_file> : Save output to disk.
 -p <proxy[:port]> : Use this proxy. (Default port is 1080)
 -P <proxy_username:proxy_password> : Proxy Authentication.
 -r : Don't search recursively.
 -R : Interactive recursion. (Asks for each directory)
 -S : Silent Mode. Don't show tested words. (For dumb terminals)
 -t : Don't force an ending '/' on URLs.
 -u <username:password> : HTTP Authentication.
 -v : Show also NOT_FOUND pages.
 -w : Don't stop on WARNING messages.
 -X <extensions> / -x <exts_file> : Append each word with this extensions.
 -z <milisecs> : Add a miliseconds delay to not cause excessive Flood.

======================== EXAMPLES =======================
 ./dirb http://url/directory/ (Simple Test)
 ./dirb http://url/ -X .html (Test files with '.html' extension)
 ./dirb http://url/ /usr/share/dirb/wordlists/vulns/apache.txt (Test with apache.txt wordlist)
 ./dirb https://secure_url/ (Simple Test with SSL)
```

html2dic – Generate a dictionary from HTML pages

```
root@kali:~# html2dic
Uso: ./html2dic <file>
```

gendict – Generator for custom dictionaries

```
root@kali:~# gendict
Usage: gendict -type pattern
```

```
 type: -n numeric [0-9]
    -c character [a-z]
    -C uppercase character [A-Z]
    -h hexa [0-f]
    -a alfanumeric [0-9a-z]
    -s case sensitive alfanumeric [0-9a-zA-Z]
 pattern: Must be an ascii string in which every 'X' character wildcard
      will be replaced with the incremental value.

Example: gendict -n thisword_X
 thisword_0
 thisword_1
 [...]
 thisword_9
```

dirb Usage Example

Scan the web server ***(http://192.168.1.224/)*** for directories using a dictionary file ***(/usr/share/wordlists/dirb/common.txt)***:

```
root@kali:~# dirb http://192.168.1.224/ /usr/share/wordlists/dirb/common.txt

-----------------
DIRB v2.21
By The Dark Raver
-----------------

START_TIME: Fri May 16 13:41:45 2014
URL_BASE: http://192.168.1.224/
WORDLIST_FILES: /usr/share/wordlists/dirb/common.txt

-----------------

GENERATED WORDS: 4592

---- Scanning URL: http://192.168.1.224/ ----
==> DIRECTORY: http://192.168.1.224/.svn/
+ http://192.168.1.224/.svn/entries (CODE:200|SIZE:2726)
+ http://192.168.1.224/cgi-bin/ (CODE:403|SIZE:1122)
==> DIRECTORY: http://192.168.1.224/config/
==> DIRECTORY: http://192.168.1.224/docs/
==> DIRECTORY: http://192.168.1.224/external/
```

Tags: enumeration http https infogathering webapps

Related Articles

WebSploit February 18, 2014

ipv6-toolkit February 18, 2014

Automater February 15, 2014