Wfuzz Usage

Wfuzz can be used for finding resources not shared (directories, servlets, scripts, etc), bruteforce GET and POST parameters for checking different kind of injections (SQL, XSS, LDAP,etc), bruteforce Forms parameters (User/Password), fuzzing etc.

Discovering directories not shared in website syntax is as follows:

Kali 2016 Console:

> wfuzz.py -z file,wordlist/general/big.txt --hc 404 http://vulnerable-web-site/FUZZ

--hc 404	:	tells wfuzz to ignore the response if the response code is 404 (Page not Found)
-z file,wordlist/general/big.txt	:	tells wfuzz to use the file common.txt as a dictionary to brute force the remote directories' name.
FUZZ	:	wherever you put this word, wfuzz will replace it with the values of the specified dictionary.

All available wordlists of wfuzz are in the following directory:

Kali 2016 Console:

> ls /usr/share/wfuzz/wordlist

Output:

general Injections others stress vulns webservices

Examples

a. Dizin Keşfetme

Hedef bir web sitesine fuzzing yaparak dizinlerini keşfedelim. Hedef web sitesi olarak tubitak'ın verdiği "My Blog" sitesini kullanalım (Not: My Blog sitesi xss_and_mysql_ file.iso sunucusunda mevcuttur. Bu sunucuyu ayağa kaldırmak için bkz. *Yaz Tatili 2014/Tubitak/Web Güveliği Ödevi/xss_and_mysql_file.iso (beni oku)).*

Hedef Web Sitesi:

http://172.16.3.60 // "My Blog"

Kali 2016 Console:

```
> wfuzz -c -z file,wordlist/general/big.txt  --hc 404 -b
  PHPSESSID=vil9comq16cq7dt7431hd7tdf7 http://172.16.3.60/FUZZ

  -c : colored output
  -b : cookie
```

Output:

```
********************************************************
* Wfuzz 2.1.3 - The Web Bruteforcer                    *
********************************************************

Target: http://172.16.3.60/FUZZ
Total requests: 3036

==================================================
ID      Response   Lines    Word       Chars       Request
==================================================

00080:  C=301      9 L         28 W        310 Ch      "admin"
00554:  C=403     10 L         30 W        287 Ch      "cgi-bin/"
00591:  C=301      9 L         28 W        312 Ch      "classes"
00712:  C=301      9 L         28 W        308 Ch      "css"
01241:  C=200     33 L         46 W        571 Ch      "header"
01313:  C=301      9 L         28 W        311 Ch      "images"
02090:  C=200     38 L         66 W        786 Ch      "post"

Total time: 8.968750
Processed Requests: 3036
Filtered Requests: 3028
Requests/sec.: 338.5086
```

Görüldüğü üzere yapılan sözlük saldırısı ile hedef web sitesinde bulunan dizinler ortaya konmuştur. Dizinlerin her biri gerçekten de hedef web sitesinde vardır. Çıktıdaki 403 yazısı görüntülenmesine izin verilmeyen dizini ifade eder. 301 yazısı yönlendirme yapan dizini ifade eder. 200 yazısı ise dizinin sorunsuz görüntülendiğini ifade eder.

* Http 403 Forbidden Uyarısı
* Http 301 Moved Permanently Uyarısı
* Http 200 OK cevabı

Http 403 Forbidden Örneği

http://172.16.3.60/cgi-bin/ [enter]

Output:

Forbidden

You don't have permission to access /cgi-bin/ on this server.

Http 301 Moved Permanently Örneği

http://172.16.3.60/admin [enter]

Output:

(Redirecting to http://172.16.3.60/login.php)

Http 200 OK Örneği

http://172.16.3.60/images/ [enter]

Output:

Index of /images

name last modified size description

key.png 19-Jul-2013 05:36 612

Apache/2.2.16 (Debian) Server at 172.16.3.60 Port 80

b. SQL Zafiyeti Tespiti

Bu uygulamada hedef web sitesine fuzzing yaparak SQL zafiyeti bulmaya çalışacağız. Hedef web sitesi olarak tubitak'ın verdiği “My Blog” sitesini kullanacağız *(Not: My Blog sitesi xss_and_mysql_ file.iso sunucusunda mevcuttur. Bu sunucuyu ayağa kaldırmak için bkz. Yaz Tatili 2014/Tubitak/Web Güveliği Ödevi/xss_and_mysql_file.iso (beni oku)).*

Hedef Web Sitesi:

http://172.16.3.60 // “My Blog”

Kali 2016 Console:

```
> wfuzz -c -z file,wordlist/Injections/SQL.txt --hc 404 -b
  PHPSESSID=vil9comq16cq7dt7431hd7tdf7 http://172.16.3.60/FUZZ
```

-c : colored output
-b : cookie

Output:

```
********************************************************
* Wfuzz 2.1.3 - The Web Bruteforcer                    *
********************************************************

Target: http://172.16.3.60/FUZZ
Total requests: 125

=====================================================
ID        Response   Lines    Word      Chars       Request
=====================================================

00000:    C=200     59 L     106 W     1334 Ch     "#"
00010:    C=400     10 L      35 W      303 Ch     "<>'"%;)(&+"
00069:    C=400     10 L      35 W      303 Ch     "&lt;&gt;&quot;'%;)(&amp;+"
00074:    C=200     59 L     106 W     1334 Ch     "/"
00076:    C=200     59 L     106 W     1334 Ch     "//"

Total time: 0.480762
Processed Requests: 125
Filtered Requests: 120
Requests/sec.: 260.0039
```

Not: SQL.txt dosyası 125 satırlık bir sözlük dosyasıdır.

Not2: Çıktı anlamlandırılamamıştır. Aynı işi yapan (fuzzing ile sqli bulma işini yapan) Burpsuite için bkz. /home/hefese/Desktop/Paketleme için Gözden Geçir/Fuzzing ile Sqli Bulma (Burpsuite).docx

Fuzzing methoduyla sqli zafiyeti yoklamasında bulunabileceğimiz gibi farklı zafiyet yoklamalarında da bulunabiliriz. Örneğin;

wordlist/Injections/XSS.txt
wordlist/Injections/All_attack.txt
wordlist/vulns/sql_inj.txt

Directory Traversal zafiyeti var mı yok mu diye test etmek için kullanılabilecek sözlükler;

wordlist/vulns/dirTraversal-nix.txt
wordlist/vulns/dirTraversal-win.txt
wordlist/vulns/dirTraversal.txt

Kaynaklar

Yaz Tatili 2014/Tubitak/Web Güvenliği Eğitimi/from_sqli_to_shell.pdf
http://tools.kali.org/web-applications/wfuzz
https://pentesterlab.com/exercises/from_sqli_to_shell/course