A DETAILED GUIDE ON DIBUSTER

WWW.HACKINGARTICLES.IN
Contents

Introduction .................................................................................................................3
What is DirBuster? .........................................................................................................3
Default Mode ...............................................................................................................3
Get Request Method .....................................................................................................4
Pure Brute Force (Numeric) ............................................................................................6
Single Sweep (Non-recursive) .........................................................................................8
Targeted Start ................................................................................................................10
Blank Extensions .........................................................................................................11
Search by File Type (.txt) ...............................................................................................12
Changing the DIR List ..................................................................................................14
Following Redirects .....................................................................................................15
Attack through Proxy ....................................................................................................17
Adding File Extensions .................................................................................................20
Evading Detective Measures .........................................................................................24
Introduction

we are focusing on the transient directory using Kali Linux tool Dibuster and trying to find hidden files and directories within a web server.

What is DirBuster?

DirBuster is an application within the Kali arsenal that is designed to brute force web and application servers. The tool can brute force directories and files. The application lets users take advantage of multi-thread functionality to get things moving faster. In this article, we will give you an overview of the tool and its basic functions.

Default Mode

We start DirBuster and only input http://testphp.vulnweb.com/ in the target URL field. Leave the rest of the options as they are. DirBuster will now auto switch between HEAD and GET requests to perform a list based brute force attack.

Let’s hit Start. DirBuster gets to work and starts brute forcing and we see various files and directories popping up in the result window.
Get Request Method

We will now set DirBuster to only use the GET request method. To make things go a little faster, the thread count is set to 200 and the "Go Faster" checkbox is checked.
In the Results – Tree View we can see findings.
Pure Brute Force (Numeric)

The way DirBuster performs this step allows a lot of control over the attack process. In this set we will be using only numerals to perform a pure brute force attack. This is done by selecting "Pure Brute Force" in the scanning type option and selecting "0-9" in the charset drop-down menu. By default, the minimum and maximum character limits are set.
In the Results – Tree View we can see findings.
Single Sweep (Non-recursive)

We will now perform a single sweep brute force where the dictionary words are used only once. To achieve this, we will unselect the "Be Recursive" checkbox.
In the Results – ListView we can see findings.
Further exploring the control options provided by DirBuster, we will set it up to start looking at the "admin" directory. In the "Dir to start with" field, type "/admin" and hit start.
In the Results – Tree View we can see findings.

Blank Extensions

DirBuster can also look into directories with a blank extension. This could potentially uncover data that might be otherwise left untouched. All we do is check the "Use Blank Extension" checkbox.
We can see the processing happen and DirBuster testing to find directories with blank extensions.

Search by File Type (.txt)

We will be setting the file extension type to .txt, DirBuster will look specifically for files with a .txt extension. Type ".txt" in the File extension field and hit "Start."
We can see the processing happen and DirBuster testing to find directories with a .txt extension.
Changing the DIR List

We will now be changing the directory list in DirBuster. Options > Advanced Options > DirBuster Options > Dir list to use. Here is where we can browse and change the list to “directory-list-2.3-medium.txt”, found at /usr/share/dirbuster/wordlists/ in Kali.

![DirBuster 1.0-RC1 -Advanced Options](image)

- Allow DirBuster to check for updates
- Dir list to use: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
- File extensions to test: php

We can see the word list is now set.
Following Redirects

DirBuster by default is not set to follow redirects during the attack, but we can enable this option under Options > Follow Redirects.
We can see the results in the scan information as the test progresses.

http://testphp.vulnweb.com:80/

Scan Information  Results - List View: Dirs: 5 Files: 31  Results - Tree View  Errors: 0

<table>
<thead>
<tr>
<th>Type</th>
<th>Found</th>
<th>Response</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>categories.php</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>File</td>
<td>disclaimer.php</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>File</td>
<td>cart.php</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>File</td>
<td>guestbook.php</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>Dir</td>
<td>/AX/</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>File</td>
<td>/AX/index.php</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>File</td>
<td>/login.php</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>Dir</td>
<td>/Mod_Rewrite_Shop/</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>Dir</td>
<td>/php/</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>File</td>
<td>Userinfo.php</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>Dir</td>
<td>/images/</td>
<td>200</td>
<td>154</td>
</tr>
<tr>
<td>File</td>
<td>search.php</td>
<td>200</td>
<td>196</td>
</tr>
<tr>
<td>Dir</td>
<td>/flash/</td>
<td>200</td>
<td>154</td>
</tr>
<tr>
<td>File</td>
<td>/flash/add.swf</td>
<td>200</td>
<td>17198</td>
</tr>
</tbody>
</table>

Current speed: 553 requests/sec
Average speed: (T) 459, (C) 459 requests/sec
Parse Queue Sizes: 0
Total Requests: 4139/107037
Time To Finish: 00:03:44

(Select and right click for more options)
Results in the Tree View.

**Attack through Proxy**

DirBuster can also attack using a proxy. In this scenario, we try to open a webpage at 192.168.1.108 but are denied access.

Access forbidden!

You don't have permission to access the requested directory. There is either no index document or the directory is read-protected.

If you think this is a server error, please contact the webmaster.

**Error 403**

192.168.1.108

Apache

We set the IP in DirBuster as the attack target.
Before we start the attack, we set up the proxy option under Options > Advance Options > Http Options. Here we check the "Run through a proxy" checkbox, input the IP 192.168.1.108 in the Host field, and set the port to 3129.
We can see the test showing results.
Adding File Extensions

Some file extensions are not set to be searched for in DirBuster, mostly image formats. We can add these to be searched for by navigating to Options > Advanced Options > HTML Parsing Options.
We will delete jpeg in this instance and click OK.
In the File Extension field we will type in “jpeg” to explicitly tell DirBuster to look for .jpeg format files.
We can see in the testing process, DirBuster is looking for and finding jpeg files.

<table>
<thead>
<tr>
<th>Testing for dirs in /</th>
<th>0%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Testing for files in / with extention .jpeg</td>
<td>0%</td>
</tr>
<tr>
<td>Testing for dirs in /images/</td>
<td>0%</td>
</tr>
<tr>
<td>Testing for files in /images/ with extention .jpeg</td>
<td>0%</td>
</tr>
<tr>
<td>Testing for dirs in /cgi-bin/</td>
<td>0%</td>
</tr>
<tr>
<td>Testing for files in /cgi-bin/ with extention .jpeg</td>
<td>0%</td>
</tr>
<tr>
<td>Testing for dirs in /admin/</td>
<td>0%</td>
</tr>
</tbody>
</table>

Current speed: 532 requests/sec
Average speed: (T) 410, (C) 410 requests/sec
Parse Queue Size: 0
Total Requests: 2050/2646589
Current number of running threads: 100

(Select and right click for more options)
Evading Detective Measures

Exceeding the warranted requests per second during an attack is a sure shot way to get flagged by any kind of detective measures put into place. DirBuster lets us control the requests per second to bypass this defense. Options > Advanced Options > Scan Options is where we can enable this setting.

![DirBuster 1.0-RC1 - Advanced Options](image)

We are setting Connection Time Out to 500, checking the Limit number of requests per second and setting that field to 20.
Once the test initiated, we will see the results. The scan was stopped to show the initial findings.
Once the scan is complete the actual findings can be seen.
We hope you enjoy using this tool. It is a great tool that’s a must in a pen tester’s arsenal.
JOIN OUR TRAINING PROGRAMS

BEGINNER

- Ethical Hacking
- Network Pentest
- Bug Bounty
- Wireless Pentest
- Network Security Essentials

ADVANCED

- Burp Suite Pro
- Web Services-API
- Pro Infrastructure VAPT
- Computer Forensics
- Android Pentest
- Advanced Metasploit
- CTF

EXPERT

- Red Team Operation
- Privilege Escalation
- APT’s - MITRE Attack Tactics
- Active Directory Attack
- MSSQL Security Assessment
- Windows
- Linux