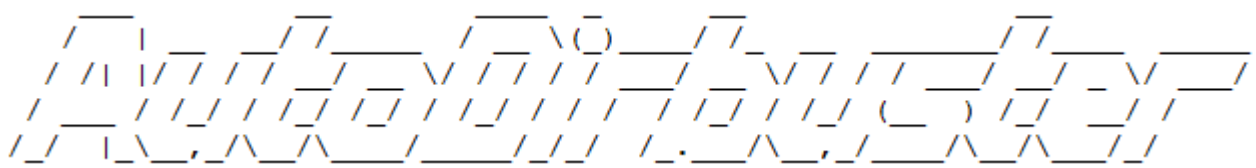
AutoDirbuster – Automatically Run and Save DirBuster Scans for Multiple IPs



If you've used OWASP's DirBuster, you know it's a great directory buster. Its speed and reliability make it one of the best directory busters currently available. However, it has one big limitation: it can only scan one target at a time.

This is fine if you're only attacking one target, but if you are attacking an entire network, then directory busting becomes a very manual process with a lot of downtime between scans. AutoDirbuster attempts to automate that process and eliminate downtime between scans.

For those who just want the code, it can be downloaded from https://github.com/NetSPI/AutoDirbuster

How does it work?

AutoDirbuster is essentially a Python wrapper for launching DirBuster. The user provides a list of targets, denoted as "IP:port" and AutoDirbuster automatically launches DirBuster for each target. However, AutoDirbuster does additional checks to ensure that the proper target is passed to DirBuster.

The workflow is as follows:

- A list of targets is provided
- A TCP connect scan is done on the target port to test if it's open
- If it's open, HTTP and HTTPS requests are sent to determine if the service is HTTP-based and whether it requires TLS
- If the service is HTTP, a check is done to determine if a previous report file is in the same directory
- Dirbuster is run using Python's subprocess.Popen(). If a timeout is specified, then after the timeout period, a SIGINT signal is sent to Dirbuster so it can safely shut down and write results to disk. A note is added to the report indicating that the scan timed out.
- The next IP:port goes through the same process (TCP connect, HTTP service query, dirbust)

What's really useful about this workflow is that a target with a closed port or non-HTTP based services running can still be provided to AutoDirbuster. The advantage of this is that Nmap scan results can be directly provided to AutoDirbuster. In fact, there's an option just for that: provide an Nmap Gnmap results file as a list of targets.

```
root@kali:~/AutoDirbuster# ./AutoDirbuster.py targets.txt -to 1 --dns
[1/3] example.com:80
Starting OWASP DirBuster 0.12 in headless mode
Starting dir/file list based brute forcing
Dir found: / - 200

Caught exit of DirBuster
Writing report
Report saved to DirBuster-Report-example.com-80.txt
Enjoy the rest of your day


[2/3] localhost:22
Target not online or service not HTTP


[3/3] localhost:80
Starting OWASP DirBuster 0.12 in headless mode
Starting dir/file list based brute forcing
Dir found: / - 200
Dir found: /tools/ - 403
Dir found: /lists/ - 403
Dir found: /javascript/ - 403
Dir found: /icons/ - 403
Dir found: /xss/ - 403
Dir found: /payload/ - 403
[!] Timeout value of 1 minutes reached, killing scan

Caught exit of DirBuster
Writing report
Report saved to DirBuster-Report-localhost-80.txt
Enjoy the rest of your day
```

Installation

The installation process is straightforward:

1. Clone the repository with git
2. Navigate to the repository on your machine and install dependencies
3. Run AutoDirbuster. If you see the usage output, the installation was a success and you're ready to use AutoDirbuster.

Copy and paste the commands below to install AutoDirbuster:

```
git clone https://github.com/NetSPI/AutoDirbuster.git
cd AutoDirbuster && pip3 install -r requirements.txt
python AutoDirbuster.py
```

If the script isn't working as intended, check the GitHub repository for common issues here.

Features

A number of features were added to make AutoDirbuster customizable.

These features include:

- **Target timeout**
  - Automatically end a scan after a given amount of time
  - Useful for targets that respond with the same status code for every request or for an unresponsive or slow target
- **Automatic DNS reverse lookup**
  - The reverse lookup hostname result will be used instead of just the IP
  - Useful for targets that are using virtual hosting
- **Gnmap mode**
  - Directly provide an Nmap Gnmap results file as the list of targets
  - Port scan and then immediately start directory busting
- **Custom wordlist**
  - AutoDirbuster uses OWASP's directory-list-2.3-small.txt by default but any list can be used
- **Single target mode**
  - Quickly launch DirBuster from the terminal against a single target without having to spend time configuring its parameters
- **Recursive mode**
- **Custom file extension list**
- **Number of connection threads**
- **Start point of the scan**

```
root@kali:~/AutoDirbuster# ./AutoDirbuster.py
usage:
    ___         __        ____  _      __               __
   /   | __  __/ /_____  / __ \(_)____/ /_  __  _______/ /____  _____
  / /| |/ / / / __/ __ \/ / / / / ___/ __ \/ / / / ___/ __/ _ \/ ___/
 / ___ / /_/ / /_/ /_/ / /_/ / / /  / /_/ / /_/ (__  ) /_/  __/ /
/_/  |_\__,_/\__/\____/_____/_/_/  /_.___/\__,_/____/\__/\___/_/

./AutoDirbuster.py [options] {target file}
    Automatically run and save Dirbuster scans for multiple IPs

Positional arguments:
    {target} Target file; list of IP:port, one per line

Optional arguments:
    Common Options:
    -g         Gnmap mode; provide a Nmap .gnmap file instead of an IP:port file
                   as a positional argument
    -st        Single target mode, positional argument is target in IP:port format
    -to        Set a timeout value in minutes for each host; default is None
    -v         Verbose mode; print service query status updates
    -f         Force mode; don't check if DirBuster report file exists, this will
                   result in previous reports being overwritten
    -h         Print this help message
    --dns      Automatically resolve IP address to hostname to use during dirbust

    Dirbuster Options:
    -d         Full path of directory that contains DirBuster.jar; default is
                   /root/AutoDirbuster/DirBuster/
    -l         Wordlist to use for list based brute force; default is OWASP's
                   directory-list-2.3-small.txt
    -e         File Extension list (e.g.: "asp,aspx"); default is None
    -t         Number of connection threads to use; default is 350
    -r         Recursive mode; default is False
    -s         Start point of the scan; default is "/"

Examples:
    python AutoDirbuster.py ip_port_list.txt
    python AutoDirbuster.py -g Nmap_results.gnmap -to 15
    python AutoDirbuster.py -g Nmap_results.gnmap -r -e "php,html" --dns
```

Recommended Workflow

- Run Nmap and find open ports, outputting the results with “-oG” or “-oA”
- Run AutoDirbuster in a terminal multiplexer, such as tmux, with the Nmap results and a timeout
  - Example: python AutoDirbuster.py -g Nmap_results.gnmap -to 15
- As the pentest progresses, periodically review the dirbust results using the included DirBuster pretty printing script dirbust_read.py, which will ignore all DirBuster error lines and only print the found directories and files

Conclusion

Directory busting is an important part of a penetration test but can be a painful manual process on its own. Using AutoDirbuster makes directory busting painless, efficient, and very fast. Give it a shot and

see if you find it useful.

https://github.com/NetSPI/AutoDirbuster