

Forum:[Hack Forums](#)

Thread: [Quick Tutorial: Basic Web Reconnaissance Using dirsearch](#)

psycheuser - August 26, 2026, 3:56 pm

If you are just starting out with web penetration testing, one of the most important first steps is discovering hidden directories and administrative panels that aren't linked on the main site.

Here is a quick, no-nonsense way to use dirsearch for basic directory enumeration:

Install and Update: Make sure you have it installed (you can grab it via git or pip), and keep a solid wordlist handy, like directory-list-2.3-small.txt.

Run the Scan: Open your terminal and run a basic command like this:

```
python3 dirsearch.py -u [http://target.com](http://target.com) -e php,html,txt
```

Filter and Analyze: Look out for status codes like 200 OK for accessible files, 301/302 for redirects, or 403 Forbidden which might indicate sensitive directories worth investigating further.

It's a straightforward technique that lays the foundation for any good assessment. For the beginners here, what tools do you usually use for your initial recon? Let's discuss below.

averi - August 28, 2026, 11:48 am

thanks

arleyne1 - August 28, 2026, 6:42 pm

(Aug 26, 2026 03:56 PM)psycheuser Wrote: [\[CITE\]](#) If you are just starting out with web penetration testing, one of the most important first steps is discovering hidden directories and administrative panels that aren't linked on the main site.

Here is a quick, no-nonsense way to use dirsearch for basic directory enumeration:

Install and Update: Make sure you have it installed (you can grab it via git or pip), and keep a solid wordlist handy, like directory-list-2.3-small.txt.

Run the Scan: Open your terminal and run a basic command like this:

```
python3 dirsearch.py -u [http://target.com](http://target.com) -e php,html,txt
```

Filter and Analyze: Look out for status codes like 200 OK for accessible files, 301/302 for redirects, or 403 Forbidden which might indicate sensitive directories worth investigating further.

It's a straightforward technique that lays the foundation for any good assessment. For the beginners here, what tools do you usually use for your initial recon? Let's discuss below.

thank you, for sharing

psycheuser - August 29, 2026, 5:07 am

(Aug 28, 2026 06:42 PM)arleyne1 Wrote: [\[CITE\]](#) thank you, for sharing

Anytime, glad it helps you out. What wordlists or tools do you normally prefer for your own runs?

(Aug 28, 2026 11:48 AM)averi Wrote: [\[CITE\]](#) thanks

Glad it helps out.

arleyne1 - August 29, 2026, 8:08 am

Dirsearch is definitely a solid one. I usually bounce between dirsearch, ffuf and gobuster depending on what I'm doing. SecLists and the old directory-list-2.3 lists are still my usual go-to wordlists too.

psycheuser - August 29, 2026, 8:19 am

(Aug 29, 2026 08:08 AM)arleyne1 Wrote: [\[CITE\]](#) Dirsearch is definitely a solid one. I usually bounce between dirsearch, ffuf and gobuster depending on what I'm doing. SecLists and the old directory-list-2.3 lists are still my usual go-to wordlists too.

Solid choices right there. ffuf speed is great when you want to run rapid fuzzing, and SecLists is a constant plug for any serious enumeration work.

Do you ever customize your wordlists by filtering out noise extensions, or do you just run directly with the standard lists?

arleyne1 - August 29, 2026, 2:44 pm

Still pretty new to it so I mostly just run the standard lists for now and see what comes back. Haven't really got into making my own custom lists yet, still learning the ropes.

psycheuser - August 29, 2026, 4:04 pm

(Aug 29, 2026 02:44 PM)arleyne1 Wrote: [\[CITE\]](#) Still pretty new to it so I mostly just run the standard lists for now and see what comes back. Haven't really got into making my own custom lists yet, still learning the ropes.

Appreciate the honesty. Starting with standard lists is the best way to get a feel for how different applications throw back codes before you start dealing with custom mutations.

When you do decide to branch out, trimming your wordlists to match the specific tech stack—like focusing strictly on .php or .aspx extensions depending on the target—saves a ton of time. Have you tried spinning up a local lab environment with Docker to run your test scans against, or are you mostly testing against live practice platforms right now?