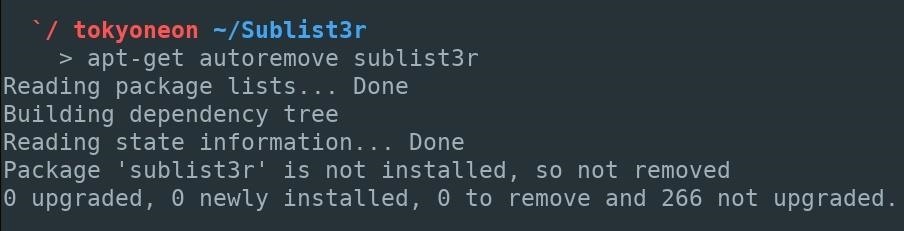
Install Sublist3r

An older version of Sublist3r can be found in the [Kali](https://tag.wonderhowto.com/how-to-do-kali/) [repositories](https://tools.kali.org/information-gathering/sublist3r). To avoid potential confusion, any version of Sublist3r which may already be installed should be removed using the below command.

[apt-get](https://null-byte.wonderhowto.com/how-to/hack-like-pro-linux-basics-for-aspiring-hacker-part-5-installing-new-software-0147591/) autoremove sublist3r

[](https://img.wonderhowto.com/img/original/07/26/63660359927135/0/636603599271350726.jpg)

Next, clone the [Sublist3r GitHub repository](https://github.com/aboul3la/Sublist3r). This will ensure the latest version is downloaded and all of Sublist3r's newest features and engines are available.

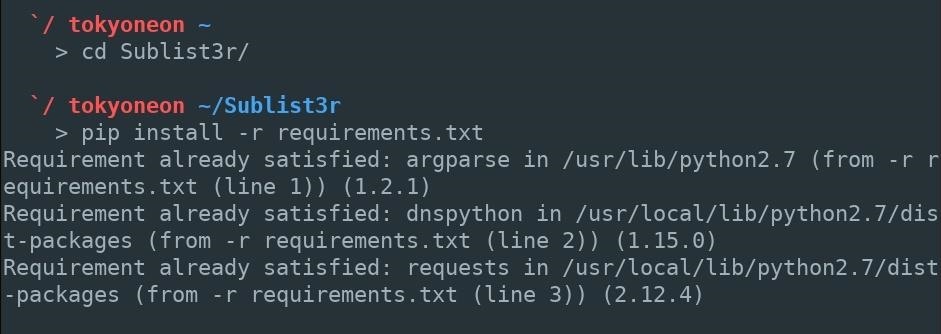
git clone https://github.com/aboul3la/Sublist3r

[](https://img.wonderhowto.com/img/original/55/87/63660359948869/0/636603599488695587.jpg)

Step 2Install Sublist3r's Dependencies

Then, [**cd**](https://null-byte.wonderhowto.com/how-to/hack-like-pro-linux-basics-for-aspiring-hacker-part-2-creating-directories-files-0147234/) into the "Sublist3r" directory and use **pip**, a tool for installing and managing Python packages, to install the Sublist3r dependencies. This command is required to run Sublist3r. The **-r** argument instructs pip to install the dependencies found in the "requirements.txt" file.

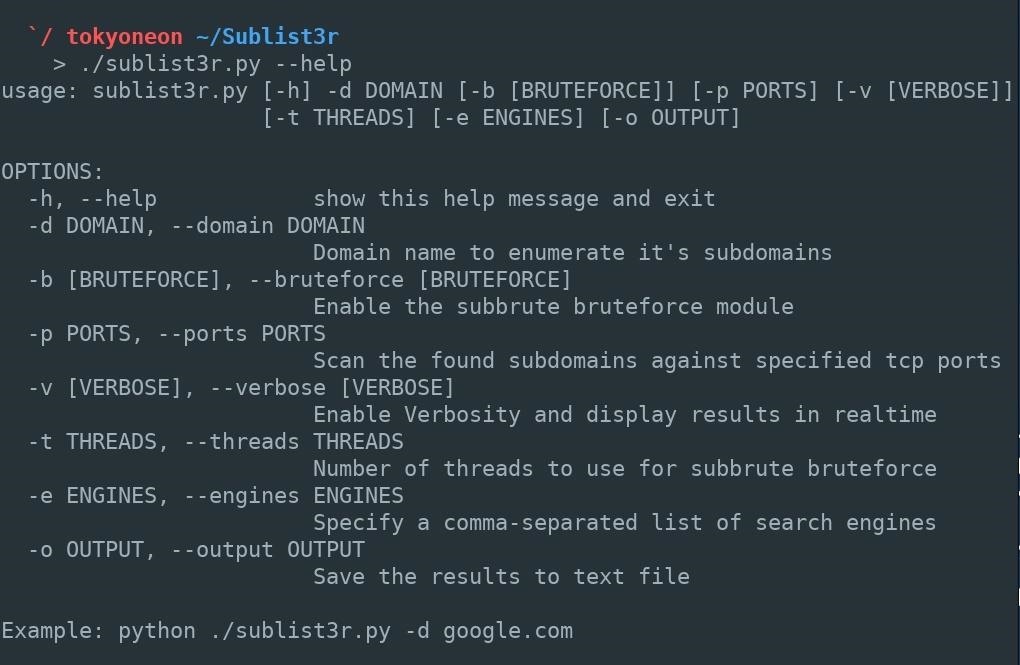
cd Sublist3r/  
pip install -r requirements.txt

[](https://img.wonderhowto.com/img/original/71/93/63660359980150/0/636603599801507193.jpg)

Step 3Verify It's Working

The **--help** (or **-h**) argument can be used to verify Sublist3r is working properly and view the available options. As you'll see, each option has a long and short form argument available.

./sublist3r.py --help

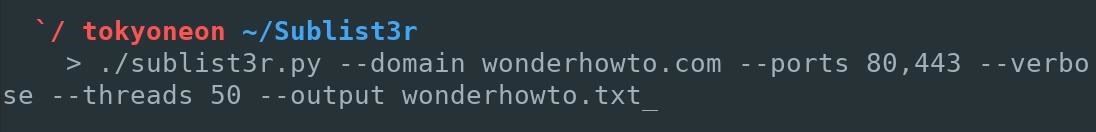
[](https://img.wonderhowto.com/img/original/79/06/63660360002900/0/636603600029007906.jpg)

That's it for cloning Sublist3r and installing dependencies — there are no modifications or configurations required.

Step 4Find Subdomains with Sublist3r

To start enumerating subdomains, use the below command.

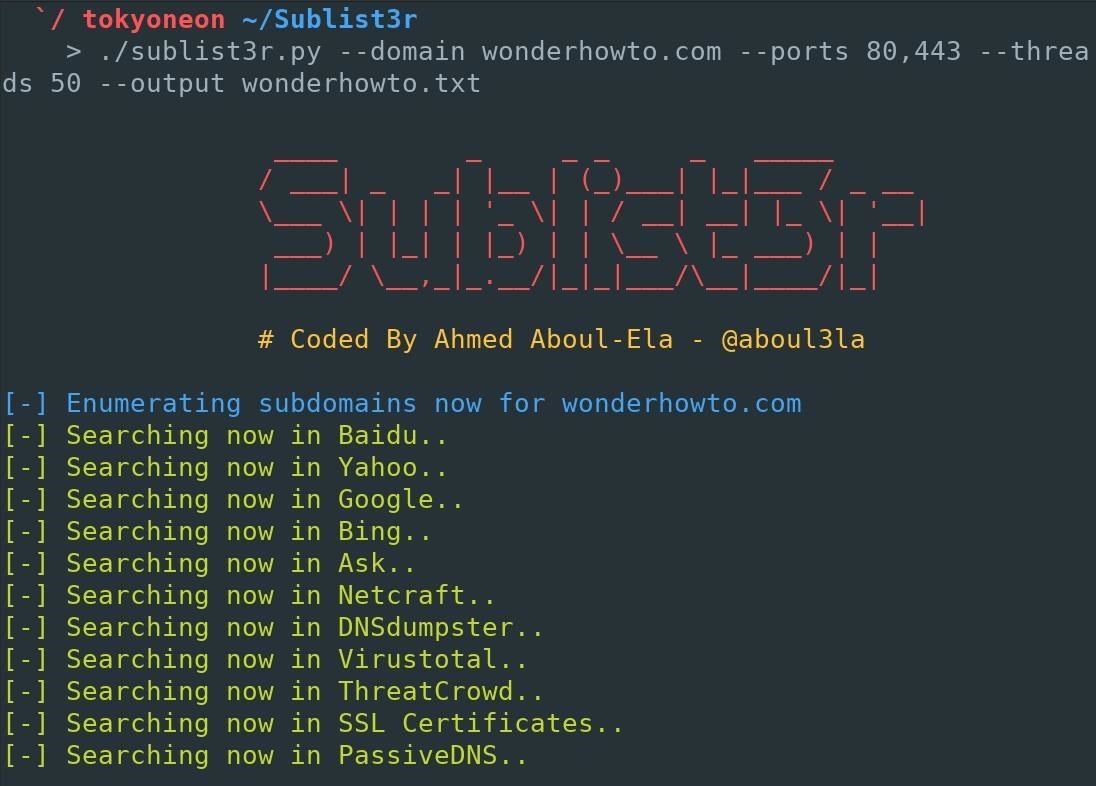
./sublist3r.py --domain target\_website.com --ports 80,443 --verbose --threads 50 --output /path/to/filename.txt

[](https://img.wonderhowto.com/img/original/73/87/63660360027400/0/636603600274007387.jpg)

* The **--domain** (or **-d**) argument defines the target website. This is the target website which Sublist3r will attempt to find subdomains for.
* Ports **80** and **443** (defined using **--ports** or **-p**) are the [official](https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers) TCP ports reserved for websites. Sublist3r can attempt to enumerate subdomains on additional ports, but as these are the most common ports used by websites on the entire internet, it's safe to restrict the command to just these ports.
* Sublist3r can print additional information related to discovered subdomains as the command is running. This is an optional setting and enabled using the **--verbose** (or **-v**) argument.
* Websites (e.g., Google) may blacklist or rate-limit requests originating from a users IP address, which would cause Sublist3r to produce fewer results. By default, Sublist3r locates subdomains within seconds. Modifying the **--thread** (or **-t**) count is another optional argument, but some readers may wish to increase (or decrease) the speed of the enumeration.
* Where the list of discovered subdomains are saved to is defined by the **--output**(or **-o**) argument. If this argument is not included in the command, Sublist3r will simply print the subdomains in the terminal.

For example, I chose the following for my query, and as seen in the image below, Sublist3r will use every search engine and database at its disposal, then print the results it finds to the .txt file we indicated. If there is no path indicated before the filename you select, the file will be created and saved in the current directory.

./sublist3r.py --domain target\_website.com --ports 80,443 --verbose --threads 50 --output filename.txt

[](https://img.wonderhowto.com/img/original/16/49/63660360075510/0/636603600755101649.jpg)

If only one or two search engines are desired, the **--engines** (or **-e**) argument can be used. The engines should be separated by a single comma (,) as shown in the below command.

./sublist3r.py -d target\_website.com --engines *engine*,*engine*,*engine* -p 80,443

Below is an example command where I enumerated WonderHowTo subdomains using only the DNSdumpster and Yahoo engines and omitted the **--output** argument.

./sublist3r.py -d wonderhowto.com --engines dnsdumpster,yahoo -p 80,443