**What?:**

Beta is a HTTP Post Request Brute forcer, that recons a site page via a get request, and then builds a post request from supplied user input.

**Usage and help:**

beta.py [-h] [-target TARGETS [TARGETS ...]]

[-user USERNAME [USERNAME ...]] [-pass PASSWORD [PASSWORD ...]]

[-target\_file TARGETFILE] [-user\_file USERNAMEFILE]

[-pass\_file PASSWORDFILE]

An inherently broken HTTP POST Request Brute forcer.

optional arguments:

-h, --help show this help message and exit

-target TARGETS [TARGETS ...], --targets TARGETS [TARGETS ...]

Please specify your target URL, for multiple use a

space delimitation.

-user USERNAME [USERNAME ...], --username USERNAME [USERNAME ...]

Specify usernames for logging into the site. For

multiple use space delimitation

-pass PASSWORD [PASSWORD ...], --password PASSWORD [PASSWORD ...]

Specify passwords for logging into the site. For

multiple use space delimitation

-target\_file TARGETFILE, --targetfile TARGETFILE

Specify the filepath to the target file. It must

contain url's of the target site, one per line

-user\_file USERNAMEFILE, --usernamefile USERNAMEFILE

Specify the filepath to the user file. It must contain

emails or username, one per line

-pass\_file PASSWORDFILE, --passwordfile PASSWORDFILE

Specify the filepath to the password file. It must

contain passwords, one per line

Common usage is beta.py -target -user -pass or beta.py -user\_file -target\_file

-pass\_file

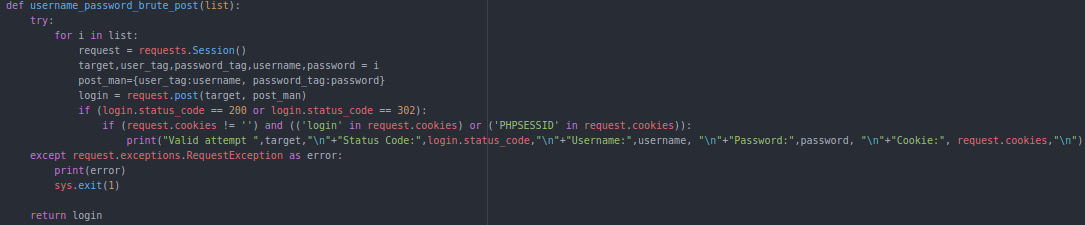
Basic HTTP Post Brute Forcing.

**Interesting functions and unpacking:**

Please review the following functions within the beta.py

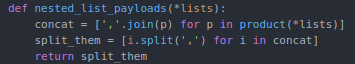
**def username\_password\_brute\_post(list)**

* **Why?** Builds a dictionary from a supplied nested llist, each nested element is unpacked within the dictionary.



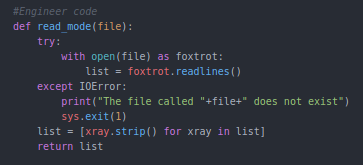
**def nested\_list\_payloads(\*lists)**

* **Why?** Builds a nested list of lists from the supplied flattened lists only.

****

**def read\_mode(file)**

* **Why?** Builds a flattened list from a file.

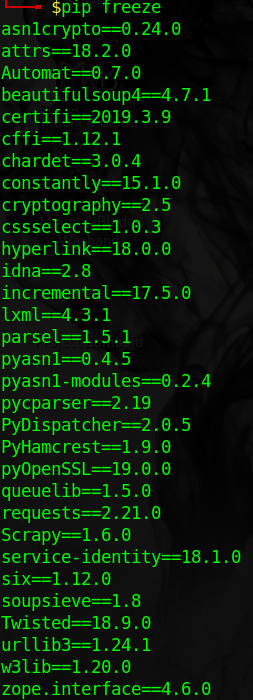


**def input\_finder(html) and def input\_tag\_get(targets)**

* **Why?** Finds input tags for users and passwords within a given page.



**Requirements:**



Please do pip install of the following dependencies (requests, cssselect, lxml)

**Default Output:**

