NAME

auto_osint_py - automate external scanning for red teams

SUMMARY

```
usage: autosint [-h] [--modules MODULES] [--timeout TIMEOUT] [--input INPUT]
[--infile INFILE] [--stdout] [-o OUTPUT_DIR] [--mode MODE]

automate osint activity
optional arguments:
 -h, --help         show this help message and exit
 --modules MODULES  domain, ip, debug
 --timeout TIMEOUT  time to wait before terminating subprocess (default=0)
 --input INPUT      comma seperated list of domains and/or ip addresses
 --infile INFILE    file containing newline seperated list of domains and/or ip
addresses
 --stdout           write output to stdout
 -o OUTPUT\_DIR      output directory. by default writes output to current
directory
 --mode MODE        LIMITED OR UNLIMITED for API calls
```

SYNOPSIS

auto_osint_py --modules [MODULES] --input / --infile [INPUTS]

DESCRIPTION

auto_osint_py is an OSINT automation script to assist in external scans for red teams by summarizing information on domains, IP addresses, web-servers and email accounts / personnel

The input to the tool is a comma seperated list of domains and/or IP addresses and the output is a directory containing subdirectories for each module specified.

MODULES

domain

Gather subdomains of a given root domain.
Inputs can be specified via the `--input` argument which accept a comma seperated list of domains or via the `--infile` argument which accepts the path of a file containing a newline seperated list of domains.

Example

```
auto_osint_py --modules domain --input example.com,test.org -o ~/scans
auto_osint_py --modules domain --infile domains.txt -o ~/scans
```

ip

Perform TCP scans of the specified input IP addresses and (if specified along with the domain module) of the IP addresses associated with the subdomains found by the `domain` module.

Example

```
auto_osint_py --modules ip --input 192.168.1.21,127.0.0.1
auto_osint_py --modules domain,ip --input example.com,test.org
```

web

This module must be run in conjunction with the `ip` module. The `web` module looks for web-servers running on the the specified input IP addresses and performs the following scans on them:-

- Sslscan if ssl is found
- Identify backend technology using whatweb
- Unlisted directories with Gobuster

email

The email module uses the HunterIO and EmailRep API to find email addresses associated with a particular domain and provides additional information like data sources, social media accounts and records of data breaches.

#### Example

```
auto_osint_py --modules email --input example.com
auto_osint_py --modules domain,email --input example.com
```

ARGUMENTS

`--modules`

Type: Required
Accepts a comma seperated list of modules to be run.
Modules include `domain,ip,web,email`
Optionally, `all` can be specified to run all modules.
#####Example

```
--modules all
--modules domain,ip,web,email
```

`--input`

Type: Required
Accepts a comma seperated list of domains and IP addresses
#####Example

```
--input example.com,192.168.1.21
```

`--infile`

Type: Required
Accepts a path to a file containing a newline seperated list of domains and IP addresses.
#####Example

```
--infile input.txt
cat input.txt
example.com
192.168.1.21
```

`--stdout`

Type: Optional
If specified, prints output to stdout in addition to saving it in a file.
#####Example

```
auto_osint_py \[ARGS\] --stdout
```

`--timeout`

Type: Optional
Specifies timeout (in seconds) for the tool. If any subprocess takes longer than timeout, it is terminated.
#####Example

```
--timeout 100
```

`-o`

Type: Optional
Specified output directory.
#####Example

```
-o /tmp/output
```

CONFIG

api.keys

The api.keys file contains the API keys required for the tool to make calls to the API service.

Syntax

```
cat api.keys
hunterio:<API KEY>
emailrep:<API KEY>
```