Hystrix-Box

Release 0.1

Apr 21, 2020

Contents

CHAPTER 1

Index

1.1 Getting Started

1.1.1 Dependencies needed for installation

- Python 3.6 or above
- **Pillow**, **console_menu** and **requests**

1.1.2 Quick install

Run the following command:

```
$ pip install hystrix-box
```

Hystrix-Box Tutorial

Note: This tutorial only cover the basic tools of `Hystrix-Box`

Install and run Hystrix-Box

Install dependencies

Hystrix-Box supports Python 3.6 and above

To check whether you have an appropriate version of Python 3:

```
$ python3 --version
```

If this does not return a version number or returns a version lower than 3.5, you will need to install Python 3

Install Hystrix-Box

Use pip, which is packaged with Python, to install Hystrix-Box and its dependencies:

```
$ pip install hystrix-box
```

Add API keys

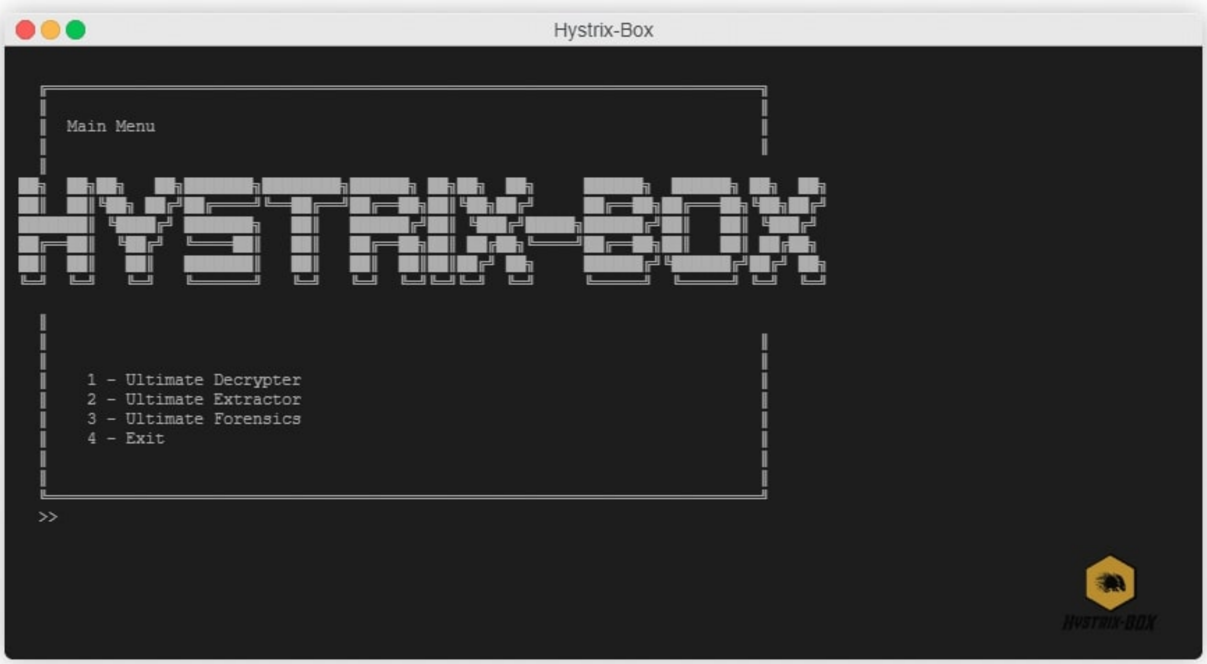
Open *HystrixBox/keys.py* and change the vars values corsponed to your API keys
you got when registering to Oxford Dictionaries

Run Hystrix-Box

Open CMD and type:

```
$ Hystrix-Box
```

If everything worked, you should be see this:



Important:

In order to use Word analysis you need to add you keys to keys.py, further information
Can be found *Add API keys*

1.2 Usage

1.2.1 Decrypter Module

Ultimate Decrypter, decrypt the given ciphertext by all known decoders. Evaluate each result and return the best result.

Usage

`[-h] (-c CIPHERTEXT | -f FILENAME) [--version] [-s DECODER] [-cl] [-cw] [-cf FORMAT] [-n NUMBER] [-v] [-o FILENAME]`

Arguments

Positional Arguments

-f Filename

Type *str*

Aliases `--filename`

Explenation Set path for file to be decrypted

or

-c Ciphertext

Type *str*

Aliases `--ciphertext`

Explenation Paste ciphertext

Optional Arguments

-h

Type *flag*

Aliases `--help`

Expleneation Show help message and exit

–version

Type *flag*

Expleneation Show the version of the tool

-d

Type *str*

Options ascii, base64, caesar, reverse, hash

Aliases `--decoder`

Expleneation Use specific decoder

-n Number

Type *int*

Default 1

Aliases `--number`

Expleneation Number of results to be printed (sorted by descending score)

Note: If none from below is selected, the script uses all the evaluators together

-cl

Type *flag*

Aliases `--checkLetter`

Expleneation Use letter analysis to evaluate the results

-cw

Type *flag*

Aliases `--checkWord`

Expleneation Use word analysis to evaluate the results

-cf Format

Type *str*

Aliases `--checkLetter`

Expleneation Search the CTF flag to evaluate the results

Important: If the CTF normal flag is omryCTF2020{XXXXXXXXXXXXXXX}

you need to enter: `omryCTF2020{}`

-v

Type *flag*

Aliases `--verbose`

Expleneation Verbose mode, printing additional information

-o Filename

Type *str*

Aliases `--output`

Expleneation Path for the file the results will be saved in

Examples

- Using code.txt file and use all evaluators:

```
>>> -f code.txt
```

- Using code.txt file and use letter analysis evaluator:

```
>>> -f code.txt -cl
```

- Decrypt string and use Base64 decoder:

```
>>> -c VGhpcyBpcyBhbiBleGFtcGxlIQ== -d base64
```

Important: When using -c flag, when the ciphertext has whitespaces in it, the argument should be written between quotation marks "

- Decrypt string (with whitespaces in it) and use flag evaluator:

```
>>> -c "}wonK_tnoD_I{0202FTCyrmo galf ym si erehW" -cf omryCTF2020{}
```

- Using code.txt file, return the top 5 results and save it in results.txt file :

```
>>> -f code.txt -n 5 -o results.txt
```

Code

This module uses the *Decoders* Package to Decode the ciphertext.
And uses the *Evaluators* Package to evaluate each result plaintext.

1.2.2 Extractor Module

Ultimate Extractor, extract information from given data

Usage

`[-h] [--version] [-v] [-o FILENAME] [-e EXTRACTOR] filename`

Arguments

Positional Arguments

Filename

Type *str*

Aliases `--filename`

Explenation Set path for file to be read for extracting

Optional Arguments

-h

Type *flag*

Aliases `--help`

Expleneation Show help message and exit

–version

Type *flag*

Expleneation Show the version of the tool

-v

Type *flag*

Aliases `--verbose`

Expleneation Verbose mode, printing additional information

-o Filename

Type *str*

Aliases `--output`

Expleneation Path for the file the results will be saved in

-e Extractor

Type *str*

Default all

Options ascii, base64, caesar, reverse, hash

Aliases `--extractor`

Expleneation Use specific extractor

Examples

- Using data.txt file and use all extractors:

```
>>> data.txt
```

- Using data.txt file and use only email extractor:

```
>>> data.txt -e email
```

- Using data.txt file and extract only url to urls.txt file:

```
>>> code.txt -o urls.txt
```

Code

This module uses the *Extractors* Package

1.2.3 Forensics Module

The Forensics module built from bunch of tools.
(Maybe in the future there will be an automated tool - an **Ultimate tool**)

add image here

All those tools has the default args of

- `-h` = help
- `--version` = print tool version
- `-v` = verbose mode
- `-o Filename` = save the result in output file

From now and on I will not describe those flags in this section.

Tools

- *Detect file type*
- *Printable strings in file*
- *Recursive Decompression*
- *Email analyzer*

Detect file type

Determine the type of a file by his header (*magic number*)

Note: Like 'file' command in Unix

Usage

```
[-h] [--version] [-v] [-o FILENAME] filename
```

Arguments

Only regular args.

Examples

- Check 'checkme' file (png file):

```
>>> checkme
File extension: png
other extensions names:
MIME: image/png
description: Portable Network Graphics file
```

Code

This module uses *Detect file type*

Printable strings in file

Find printable strings in binary files

Note: Like 'strings' command in Unix

Usage

`[-h] [--version] [-v] [-o FILENAME] [-n NUMBER] filename`

Arguments

Optional Arguments

-n Number

Type *int*

Expleneation Print only sequences of characters that are at least min-len of this number

Default 4

Examples

- Search in 'checkme' file:

```
>>> checkme
Strings will be printed here
```

- Search in 'longwords' for strings at least 10 characters long:

```
>>> checkme -n 10
Strings will be printed here
```

Code

This module uses *Strings*

Recursive Decompression

Decompress nested zip files, saves the files hierarchy (nested zips changed to directories).

Usage

```
[-h] [--version] [-v] [-o FILENAME] [-p PATH] filename
```

Arguments

Optional Arguments

-p Path

Type *str*

Expleneation Set path to extract the zip files

Default Current directory

Examples

- Extract nested.zip file to current directory

```
>>> nested.zip
```

- Extract nested.zip file to directory named 'Data' in the current directory

```
>>> nested.zip -p Data
```

Code

This module uses *Recursive Decompression*

Email analyzer

Analyze email file headers to extract important information

- Subject
- Data
- From
- To
- Message-ID

- Unsubscribe URL
- Return Path (The email address when replying to this message)
- Content-Type (Check for attached files)
- Received (All the station the email pass through, track the sender)

Usage

```
[-h] [--version] [-v] [-o FILENAME] filename
```

Arguments

Only regular args

Examples

- Analyze email file with the name sendme.eml

```
>>> sendme.eml
Subject: Coming Wednesday, April 1st... Nailed It! Season 4
Date: Tue, 24 Mar 2020 18:02:10 +0000
From: Netflix <info@mailer.netflix.com>
To: JohnSmlth@hotmail.com
Message-ID: <010001710db57e6f-f9850d85-346e-4249-9551-71042c80ff5b-
↪000000@email.amazonses.com>
List-Unsubscribe:
↪<mailto:S0VXTkIzSUUyRkMzTkMyQ05KWktVSzZHRUFDNURF@unsubscribe.netflix.
↪com>, <https://www.netflix.com/EmailUnsubscribe?id=BQE0AAEBENqyIWqRNrh
↪%2B7
↪%2FV5HCi4iKmAgHeFZDEnVSyIsgyi0afmQoJVpj1JX60NWdqEnhgc3v9rrhZtAXKmQ753EK64gUYakH9o2rLLGZ8FC
↪%2BChey9LRE13FZe4%2Bo5C3KPNazPc%2BK7TzskJElw15&lnktrk=EMP&
↪g=6A5F6B01C976E12BCF1F50C0AA5062A9645262CD&lkid=unsubscribe_link>
Return-Path:
 010001710db57e6f-f9850d85-346e-4249-9551-71042c80ff5b-000000@mailer.
↪netflix.com
Content-Type: multipart/alternative;
    boundary="----=_Part_87580_587082350.1585072930404"
Received: from VI1EUR04HT047.eop-eur04.prod.protection.outlook.com
 (2603:10b6:208:51::34) by MN2PR05MB6573.namprd05.prod.outlook.com with␣
↪HTTPS
 via BL0PR02CA0093.NAMPRD02.PROD.OUTLOOK.COM; Tue, 24 Mar 2020 18:02:12␣
↪+0000
Received: from VI1EUR04FT022.eop-eur04.prod.protection.outlook.com
 (2a01:111:e400:7e0e::33) by
 VI1EUR04HT047.eop-eur04.prod.protection.outlook.com␣
↪(2a01:111:e400:7e0e::349)
 with Microsoft SMTP Server (version=TLS1_2,
 cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384) id 15.20.2814.13; Tue, 24␣
↪Mar
 2020 18:02:11 +0000
```

Code

This module uses *Email analyzer*

1.3 Code

1.3.1 Decoders

Decoder

class `HystrixBox.Decoders.`**`Decoder`**

Bases: `object`

A class used to represent a Decoder

static decode(*text*)

Decode the text by the cipher

If there are multiple ways to decode the text, return all of them

Parameters **text** (*str*) – The cipher-text

Returns List of the plain-texts (or plain-text) after decode

Return type list

Raises **NotImplementedError** – If the decode function not set in the decoder

classmethod safe_decode(*text*)

Validate the format of the text and decode it

First check if the text is in the format of the cipher, if so decode it. If the text is not in the format, return empty list

Parameters **text** (*str*) – The cipher-text

Returns List of the plain-texts (or plain-text) after decode

Return type list

static validate(*text*)

Validate string format for this cipher.

Parameters **text** (*str*) – The cipher-text

Returns Either the text is in the cipher format or not

Return type bool

Raises **NotImplementedError** – If the validate function not set in the decoder

Ascii Decoder

class `HystrixBox.Decoders.`**`ASCIIDecoder`**

Bases: `HystrixBox.Decoders.Decoder.Decoder`

A class used to represent a ASCII decoder

Example `84 104 105 115 32 105 115 32 97 110 32 101 120 97 109 112 108 101 33 -> This is an example!`

Base64 Decoder

class `HystrixBox.Decoders.`**`Base64Decoder`**

Bases: `HystrixBox.Decoders.Decoder.Decoder`

A class used to represent a Base64 decoder

Example `VGhpcyBpcyBhbiBleGFtcGxlIQ== -> This is an example!`

Caesar Decoder

class `HystrixBox.Decoders.`**`CaesarDecoder`**

Bases: `HystrixBox.Decoders.Decoder.Decoder`

A class used to represent a Caesar decoder

Note: brute-force all options (shifts) and return list of all possible options starting from 1-shift to 25-shift

Example `Rfgq gq yl cvyknjc! -> [Sghr hr zm dwzlokd!, This is an example!, Uijt jt bo fybnqmf!] and go on`

Hash Decoder

class `HystrixBox.Decoders.`**`HashDecoder`**

Bases: `HystrixBox.Decoders.Decoder.Decoder`

A class used to represent a Hash decoder

Note: correctly support only md5 hash using www.nitrxgen.net API

Example `a85a7dae016693c9351110c357e4b609 -> This is an example!`

Reverse Decoder

class `HystrixBox.Decoders.`**`ReverseDecoder`**

Bases: `HystrixBox.Decoders.Decoder.Decoder`

A class used to represent a Reverse decoder

Example `!elpmaxe na si siht -> This is an example!`

T9 Decoder

class `HystrixBox.Decoders.`**`T9Decoder`**

Bases: `HystrixBox.Decoders.Decoder.Decoder`

A class used to represent a T9 decoder, and old phone keypad (numbers to text)

Example `!8 44 444 7777 0 444 7777 0 2 66 0 33 99 2 6 7 555 33 -> This is an example!`

1.3.2 Extractors

Extractor

class `HystrixBox.Extractors.`**`Extractor`**

Bases: `object`

A class used to represent a Extractor

static extract (*self*, *text*)

Extract specific information from data (text->str) usually by regex

Parameters **`text`** (*`str`*) – The data to read from

Returns list of occurrences of the desired information

Return type list

Raises **`NotImplementedError`** – If the extract function not set in the extractor

Email Extractor

class `HystrixBox.Extractors.`**`EmailExtractor`**

Bases: `HystrixBox.Extractors.Extractor.Extractor`

A class used to represent an email extractor

Note: Regex according to RFC 5322

Ip Extractor

class `HystrixBox.Extractors.`**`IPExtractor`**

Bases: `HystrixBox.Extractors.Extractor.Extractor`

A class used to represent an Ip extractor

Note: Regex include both IPv4 and IPv6 formats

Md5 (Hash) Extractor

class `HystrixBox.Extractors.`**`MD5Extractor`**

Bases: `HystrixBox.Extractors.Extractor.Extractor`

A class used to represent an md5 Hash extractor

Note: Regex include both IPv4 and IPv6 formats

URL Extractor

class HystrixBox.Extractors.**URLExtractor**
Bases: HystrixBox.Extractors.Extractor.Extractor

A class used to represent an URL extractor

Note: Regex include prefix http or https, IPv4, IPv6, postfix port and resource path.

All URL with this format: `((?:https?://)?(?:STANDARD_URL|IPv4|IPv6)(?:PORT)?(?:RESSOURCE_PATH)`

1.3.3 Tools

Strings

HystrixBox.Tools.**strings**(*filename*, *minChars=4*)
Search printable strings in binary file

Parameters

- **filename** (*str*) – The file to be read
- **minChars** (*int*) – Min-len of characters to return string *(default 4)*

Returns List of printable strings

Return type list

Recursive Decompression

HystrixBox.Tools.**extract_recursive**(*filename*, *path=''*)
Decompress nested zip files

Parameters

- **filename** (*str*) – The file to be extracted
- **path** (*str*) – Path to extracted files *(default current directory)*

Returns None

Return type None

Email analyzer

HystrixBox.Tools.**email_analyzer**(*filename*)
Analyze email file headers

Parameters **filename** (*str*) – The file to analyze

Returns List of important information from the email header

Return type list

Detect file type

Core functions

`HystrixBox.Tools.fileType.`**`get_header`**(*filename*)

Extract header from file

Parameters **`filename`** (*str*) – filename to be read

Returns Header of the file

Return type bytearray

`HystrixBox.Tools.fileType.`**`getFileExtension`**(*filename*)

Detect file extension

Parameters **`filename`** (*str*) – filename to be checked

Returns extension or None if not found

Return type *Extension*

Extension

`class` `HystrixBox.Tools.fileType.`**`Extension`**(*extension*, *mime*, *description*, *otherExtensions=''*)

A class used to represent a Extension

Parameters

- **`extension`** (*str*) – Extension name
- **`otherExtensions`** (*str*) – Other possible extension names (if there are)
- **`mime`** (*str*) – MIME (Multipurpose Internet Mail Extensions)

Param description: Description on the extension

Type description: str

`check`(*header*)

Check if the correct magic numbers are in the file header

Parameters **`header`** (*str*) – Header of the file to be checked

Returns Either the file is according to the magic numbers or not

Return type bool

Raises **`NotImplementedError`** – If the check function not set in the extension

Extension inheritances

Application

- **`class`** `HystrixBox.Tools.fileType.`**`Pcap`**
- **`class`** `HystrixBox.Tools.fileType.`**`Db`**
- **`class`** `HystrixBox.Tools.fileType.`**`Pdf`**
- **`class`** `HystrixBox.Tools.fileType.`**`Exe`**

- **class** HystrixBox.Tools.fileType.**Elf**
- **class** HystrixBox.Tools.fileType.**Psd**
- **class** HystrixBox.Tools.fileType.**Flash**
- **class** HystrixBox.Tools.fileType.**Office**

Archives

- **class** HystrixBox.Tools.fileType.**Zip**
- **class** HystrixBox.Tools.fileType.**Rar**
- **class** HystrixBox.Tools.fileType.**Sevenz**
- **class** HystrixBox.Tools.fileType.**Jar**
- **class** HystrixBox.Tools.fileType.**Tarz**
- **class** HystrixBox.Tools.fileType.**Tarbz2**
- **class** HystrixBox.Tools.fileType.**Tarxz**
- **class** HystrixBox.Tools.fileType.**Tar**

Audio

- **class** HystrixBox.Tools.fileType.**Wav**
- **class** HystrixBox.Tools.fileType.**Aiff**
- **class** HystrixBox.Tools.fileType.**Mp3**
- **class** HystrixBox.Tools.fileType.**Aac**
- **class** HystrixBox.Tools.fileType.**Mid**
- **class** HystrixBox.Tools.fileType.**Flac**
- **class** HystrixBox.Tools.fileType.**M4a**
- **class** HystrixBox.Tools.fileType.**Ogg**
- **class** HystrixBox.Tools.fileType.**Amr**

Font

- **class** HystrixBox.Tools.fileType.**Otf**
- **class** HystrixBox.Tools.fileType.**Ttf**

Image

- **class** HystrixBox.Tools.fileType.**Jpeg**
- **class** HystrixBox.Tools.fileType.**Png**
- **class** HystrixBox.Tools.fileType.**Gif**
- **class** HystrixBox.Tools.fileType.**Webp**

- **class** HystrixBox.Tools.fileType.**Cr2**
- **class** HystrixBox.Tools.fileType.**Tiff**
- **class** HystrixBox.Tools.fileType.**Bmp**
- **class** HystrixBox.Tools.fileType.**Fits**
- **class** HystrixBox.Tools.fileType.**Ico**

Video

- **class** HystrixBox.Tools.fileType.**Flv**
- **class** HystrixBox.Tools.fileType.**Matroska**
- **class** HystrixBox.Tools.fileType.**Avi**
- **class** HystrixBox.Tools.fileType.**Mp4**
- **class** HystrixBox.Tools.fileType.**Mov**
- **class** HystrixBox.Tools.fileType.**Wmv**

1.3.4 Evaluators

Evaluator

Flag evaluator

Letter analysis evaluator

Word analysis evaluator

1.4 Support

If you have any problems or questions about Hystrix-Box, you are invited to visit any of the following support channels, where our staff will be happy to assist.

Please respect our time and effort,by not asking the same question in multiple places.

1.4.1 Discord

The Hystrix-Box Discord is open to all users and developers. To join, head to: https://discord.gg/nwrZUuU

Please read the **#rules** channel first and use the proper channel for your needs

1.4.2 Issues

If you think you've found a bug, or you'd like to suggest a new feature, please check the current list at https://github.com/zomry1/Hystrix-Box/issues. If your bug or suggestion isn't there, raise a new issue, providing as much relevant context as possible.

1.5 Release notes

1.5.1 0.1

Hystrix-Box 0.1 release notes

What's new

Initialize Project

Bug fixes

- None

Index

R

S

T

U

V

W

Z