IOC python script Documentation

For

VirusTotal

& ThreatBook Intelligence

Prepared by:

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Last revision date | Author | Remarks |
| 1 | 27/07/2023 | Chin Sek Yi |  |

**TABLE OF CONTENTS:**

1. General info
2. Requirements
3. How-to-guide
4. Terminal output
5. Error

General Info:

1. VirusTotal API allowance - using public API:

* Request rate 4 lookups / min
* Daily quota 500 lookups / day
* Hourly quota 240 lookups/ hr
* Monthly quota 15.5 K lookups / month

Previous runs: 1300 lines per day

\*Even if you have many API-keys, virusTotal will limit your endpoints to ~1300 lines per day (Don’t know exactly why).

ThreatBook Community API

* Daily quota 100 lookups / day
* Reads in IP only

ThreatBook Premium API

* Unlimited quota
* Reads in IP and domain

Requirements:

These libraries are required to perform data cleaning and processing in VSCode.

Ensure that these libraries are installed in your computer.

* pandas
* numpy
* openpyxl

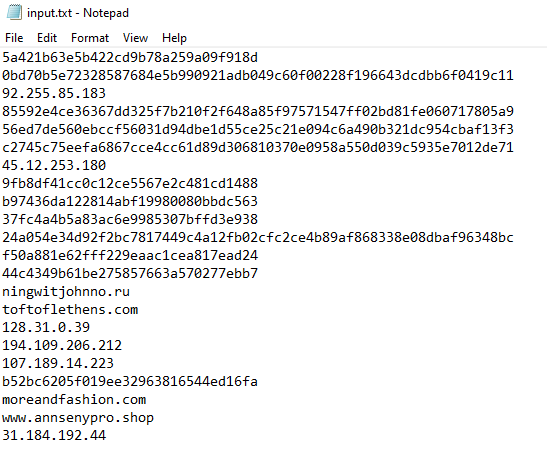
Python script required (depending on which platform you want to use):

* Threatbook\_community.py
* Threatbook\_premium.py
* virusTotal script.py

How to Guide:

1. Place your IP/domain/hash in a .txt file. Each of the input string must be in a separate line.

E.g.



2. Ensure that the python script and your source txt file are in the same working directory.

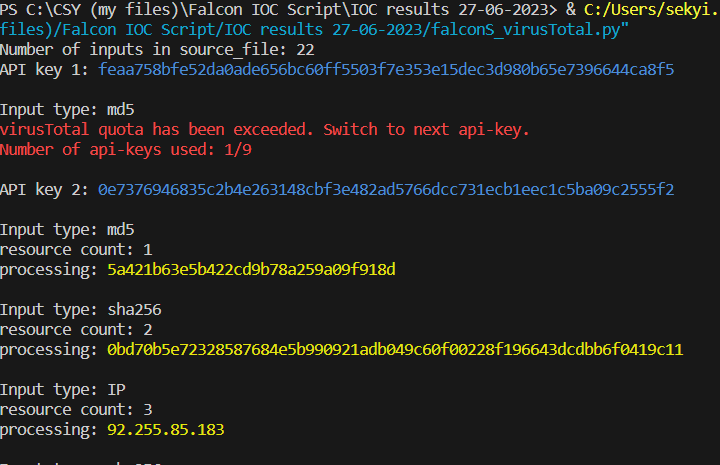
3. Open python script in VScode. Under Filename input section, change the source\_file and destination\_file name to as required. Now, run the script.

\* Ensure that your source\_fie is closed. Otherwise, you will encounter PermissionError.

4. The output will be saved in a excel file. Double check the terminal output to check if a separate batch is needed. See below.

Terminal output:

As you are running the script, you might see:



* + *API key 1: feaa758bfe52da0ade656bc60ff5503f7e353e15dec3d980b65e7396644ca8f5*

refers to the current API key that is being used. The api-key is coloured in blue.

* + *virusTotal quota has been exceeded. Switch to next api-key. Number of api-keys used: 1/9*

Tells you that the quota used by the current API key has been exceeded. In the image above, API key 1 has been used up. Hence, the number of api-keys used is one 1/9.

* + *Input type: md5*

Tells you the input type of the input currently being processed, eg IP/domain/sha256/md5

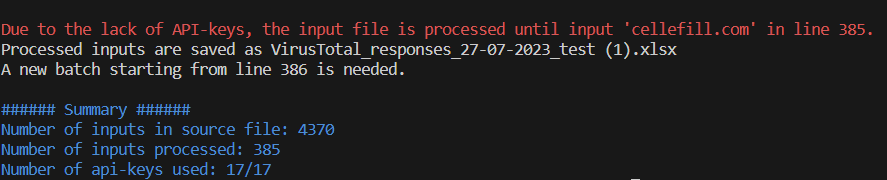
* + *Resource count: 1*

Tells you the number of inputs that have been processed so far

* + *processing: 5a421b63e5b422cd9b78a259a09f918d*

Tells you the input that is currently processed

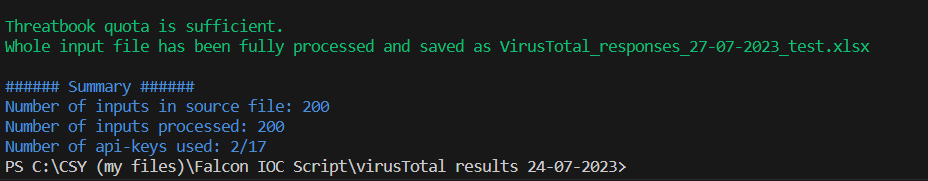
After script is ran:



For virusTotal.py, we have 17 api-keys that can be used to get the virusTotal report. The limit by virusTotal is 500 lookups per day and 240 lookups per hour. Although we have many api-keys, virusTotal will limit your endpoint to ~1300 lines per day. Hence, a separate batch of inputs needs to be run. \*Previously, 4370 lines are ran in 4 batches.

For ThreatBook\_community.py, we have 11 api-keys. There’s a limit of about 10 lookups / minute for each api-key. Hence, we need to run the script in batches.

* For both platforms, you will see the red output above. It tells you that the input file is not fully processed. That means that only some, and not all of the inputs in your source file has been ran in virusTotal.
* The red text also tells you the line that is last processed. In the case above, the last processed line is 385. Hence, your output excel file will contain results for inputs from 1 to 385.
* Later on, after the virusTotal quota is reset, you can rerun the txt script starting from the 386st line. \*So remember to delete the first 385 lines in the txt file!



If the combined quotas of all the api-keys are sufficient, your whole input file can be processed completely and thus a separate batch is not needed. The green text above will be shown.

Also, the number of inputs in source\_file will be equal to the number of inputs processed in the Summary.

The Summary gives you some information about the number of the inputs in the source file, the number of inputs processed, and the number of api-keys used.

Error:

* PermissionError: [Errno 13] Permission denied: filename.xlsx'

Occurs when your source or destination file is opened.