CTIZINE

VISIDATA EDITION

CONTEXT

Visidata is a Python library for data visualization and analysis that has been developed by saulpw.

In CTI field, we often relies on Excel to store and analyze data. You may encounter some issue with Excel, handling JSON or even CSV files could be tricky, writing formula is sometimes a pain, and medium sized dataset could start to slow seriously the UI. If you are more tech-savy, you will probably write plenty of python scripts and end up with a directory blowed up of CSV or JSON files named with meaningful names such as `iocs.json` or `test2.csv`. And I told you, in two weeks you may hate your past self.

Visidata excels here (you got it?)! You are going to have rapid insights on your data in a fraction of time. Visidata will open most of data format (xlsx, json, log file, csv, zip, sqlite, txt..) in your terminal and with keyboard shortcuts your can easily split columns and summarize or filter data. If you have Python basic knowledge you might even use it to transform and enrich data.

Visidata is not a new everything shiny tool but it is definitely very handy to handle files containing a handful to hundreds of thousand records.

USE CASES

- Rapidly parse log files to extract the most seen IP or path
- Search for patterns in a list of domains : stats on TLD used, domains matching a regexp, simple DGA fingerprinting, data in DNS records (hosted on same IP range, similar TXT record...)
- Search for patterns in a list of IP addresses: most seen AS number or IP range.
- Easily reformat a field containing a weird date format, or extract the day, the weekday, the week...
- Extract data from a JSON file obtained from other tools such as VirusTotal or UrlScan

INSTALLATION AND FIRST STEPS

If you already have Python installed on your device, you can install Visidata with on command, else I recommend you installing the uv utilities from their Github release page.

```
# With Python
python -m pip install visidata
# With uv installed
uv tool install visidata
```

For Windows users, you might add `.exe` to python or uv, and be informed that the terminal interface might be a bit capricious. If you want to keep your sanity, at least use Windows Terminal and not base cmd or powershell.

To open a file, simply type:

```
visidata <path_to_file>
```

SHORTCUTS

Visidata provides VIM likes bindings, there are bindings for everything but I give you the ones I use the more based on my experience.

Key	Effect	Example
`Pipe`	Select data from a pattern	`\.biz$` to select only domains under biz TLD
`\`	Deselect data from a pattern	`\.[a-z]{2}$` to filter out domains under ccTLD
`,`	Select row containing the same value	
`=`	Python expression	`len(domain_column_name)` generate a column containing the length of each domain
`:`	Split on the specified pattern	`\.` on a domain column will generate two new columns containing the domain name and the TLD
`;`	Extract group from the specified pattern	`(https?):\/\/([^\/]+)(.*)` extracts protocol, FQDN and path from an URL column

THE `~/.VISIDATARC` FILE

As visidata relies on Python, you can import an external library at the runtime to use it to manipulate your data. Visidata create a file named `.visidatarc` in your `$HOME`, you can add import or define functions in it that will be usable everything you open a new file.

For example, mine `.visidatarc` file import handy tools such as `tldextract`, `whoisit`, `dnspython` or `requests`.

Now I can type `=tldextract.parse(column_name)` in a column containing URL or a domain and BAM I have a new column containing the TLD (even if it's some weird things such as .co.uk), the registered name and the subdomain.

Same things goes with whoisit a Whois/RDAP library that gives information related to IP addresses, domain, AS number.

OTHER INTERESTING FEATURES

- **Replay**: you can save your current workflow or transform at any moment with `Ctrl+D` and then replay it later on any other file, which is useful when you do some transform from time to time but don't want to write any Python script to handle these data. It save the instructions with as a `.vdj` file that could be specified with the `-p` flag when starting visidata.
- **Merge**: if you fetch many JSON or CSV file from a source, then you can easily merge them. When opened, select several datasets (`s`) in the Sheet View (`Shift+S`) then press `&`. It will displays many way to merge the data (inner, outer, full join...)
- **Formatting**: open a JSON, press `g(`, save as a xlsx file. Congratulation, you have transformed a JSON file in a flat xlsx file which will be easier to share with your colleagues.

Key	Effect	Example
`~ or # or @`	Set column type to str/int/date	
`Shift+f`	Count occurrence	
`"`	Open selected rows in a new sheet	To save only a subset of data
`[ or ]`	Sort column ascending or descending	
`(`	Expand a list or a dict column	Useful to transform nested JSON in a flat CSV
`z+Shift+M`	Unfurl columns	Create new rows containing each value stored in a list or dict. Usefull to produce stats from fields containing tags or multiple DNS records