Automated Catalog Set Based Obfuscation

Summary

This plug-in is designed to accept catalog set IDs and obfuscate all column members of the catalog set.

Adding or Removing Catalog Set

Obtain Catalog Set ID

Catalog set ID can be obtained using the catalog set page URL.

E.g. `http://your_url.com/catalog_set/2/` then the catalog set ID is 2.

Modifying Configuration

Changing Catalog Set IDs

Find `IN_CS_IDS` and set it to a comma seprated string of catalog set IDs.

- To obfuscate all columns under catalog set #14: `IN_CS_IDS = '14'`
- To obfuscate all columns under catalog set #1, #3 and #5: `IN_CS_IDS = '1,3,5'`
- Stop and start the service using the commands in the section below

Changing Application DB Path

WARNING: Please make sure to move the file `tag_db` as well

- Create the data directory you wish to use
- Add the complete path into `DATA_DIR`

Usage

The code works as a daemon. You can manage it like most services in Linux:

- Starting: `sudo python3 <path to obfuscator.py> start`
- Stopping: `sudo python3 <path to obfuscator.py> stop`
- Status: `sudo python3 <path to obfuscator.py> status`

Install Dependencies

- Debian Linux: `sudo apt-get install python3-dev`
- RHEL Linux: `sudo yum install python3x-dev`
  - E.g. if you are running python 3.6, then `sudo yum install python36-dev`
- `sudo -H python3 -m pip install service`
- NOTE: you may need to install GCC

First Time Setup

- Unpack `config.py`, `support_funcs.py`, and `obfuscator.py` into any directory on your system
- Change the `DATA_DIR` location to the appropriate path
- Start the service `sudo python3 <path to obfuscator.py> start`