log-sensitive-data-censor

项目描述

一个简单的使用正则表达式来过滤日志中敏感信息的工具。

使用方式

直接使用命令行java -jar运行。

```
$ java -jar log-sensitive-data-censor-1.0.0.jar --help
```

使用`-h / --help`会打印参数格式。

命令参数如下：

命令	是否必须	参数描述
`-o / --out [输入文件路径名称]`	是	文件相对和绝对路径均可
`-i / --in [输出文件路径名称]`	是	文件相对和绝对路径均可
`-d / --date [扫描起始日期]`	否	日期格式为`yyyy-MM-dd`，并且日期必须在每一行的句首。
`-r / --regex [自定义名称1] [自定义正则表达式1] [自定义名称2] [自定义正则表达式2] [自定义名称3] [自定义正则表达式3] ...`	否	自定义名称和表达式中不要含有空格，需要注意表达式和名称的顺序。
`-c / --clear`	否	不使用内置正则表达式，前提是必须使用了自定义正则表达式（`-r / --regex`），否则不生效。

案例

假设使用以下命令：

```
$ java -jar log-sensitive-data-censor-1.0.0.jar \
        -i C:\Users\Administrator\Documents\test-in.log \
        -o C:\Users\Administrator\Documents\test-out.log \
        -d 2022-05-19 \
        -r 测试 2022-05-19\s14
```

命令表示：

1. 输入文件[`C:\Users\Administrator\Documents\test-in.log`]进行过滤。
2. 输出过滤后文件[`C:\Users\Administrator\Documents\test-out.log`]。

3. 从日期[`2022-05-19`]开始扫描。
4. 添加自定义名称为[`测试`]的正则表达式[`2022-05-19\s14`]进行扫描

之后开始执行命令：

```
文件[test-in.log]大小为199171853字节(189.95MB)
当前进度：99%
完成
```

执行完之后输出的文件是这样的：

```
行号：1761，可能存在手机号，身份证号，银行账号，测试：
手机号：
"},"REMARK":"xxxx","xxxx_PHONE":"15640141998","xxxx_ID":"
","xxxx_PHONE":"15640141998","xxxx_ID":null,"xxxx_ID":"
身份证号：
","xxxxx_NAME":"苏*","xxxxx_NAME":"damao_huntun","xxxxx_CODES":"
[xxxx]","ID_NO":"11000000000000000","xxxx_CODE":"
银行账号：
","xxxx_NAME":"xxxxx","xxxx_NAME":"xxxx","xxxx_CODES":"
[xxxx]","BANK_CARD_NUM":"500233199605084428","xxxx_CODE":"
测试：
2022-05-19 14:00:05.216||ERROR||xxxx||xxxx

行号：1761，可能存在手机号，测试：
手机号：
"},"REMARK":"xxxx","xxxx_PHONE":"15640141998","xxxx_ID":"
","xxxx_PHONE":"15640141998","xxxx_ID":null,"xxxx_ID":"
测试：
2022-05-19 14:00:05.216||ERROR||xxxx||xxxx
```

输出的文件中带有原文件的行号，方便进行查阅。

说明

- 脚本目前还是单线程，未来会优化多线程。
- 脚本对内存使用影响甚微，依赖CPU的性能。