Network Administration Practice Homework 1: Python Scripts

weicc & blzhuang

Requirements

- 1-1 Web crawler (45%+5%)
  - Entry point: `nahw1-1_{student_ID}.py`
  - Login options
    a) **NCTU Portal (5% Bonus)**
    b) **Course Registration System**
- 1-2 Auth Log parser – Count ssh login failed (45%)
  - Entry point: `nahw1-2_{student_ID}.py`
- Readme file (Readme.md) (10%)
  - List used Package / library for each part
  - Login by NCTU Portal or Course Selection System
- Due data: 2018/03/22 23:59
  - Upload `nahw1-{$student_ID}-{$YYYYMMDD-HHMM}.tar` on New E3 (https://e3new.nctu.edu.tw)

Readme File: Sample

```
# Network Adminstration - Homework 1

## Part I - Web crawler

Login to NCTU portal and get schedule on course selection system.

### Dependency
- Pillow
- pytesseract
- BeautifulSoup
- PrettyTable

## Part II - Auth log parser

Parse auth.log file.

## Dependency
.....
```

1-1: Web Crawler – Requirements (1/2)

- Input format
  - `python script_name username`
- Argument parser 5%
  - -h show usage
- Captcha Recognize 20%
  - Login by course selection system
  - Login by NCTU Portal
- Parse HTML & Print Table (one of following)
  - Can parse schedule as list from HTML 10%
  - Can parse schedule as list from HTML and print as table 20%
- Bonus 5% - Login by NCTU Portal
  - Solve all kinds of captcha
  - Relay login session from NCTU Portal to course selection system

1-1: Web Crawler – Requirements (2/2)

- ❑ Only Python script allowed
  - Call shell script is not allowed
  - Call NodeJS script is not allowed
  - `subprocess` not allowed
  - `os.system` not allowed
- ❑ Python 3 only

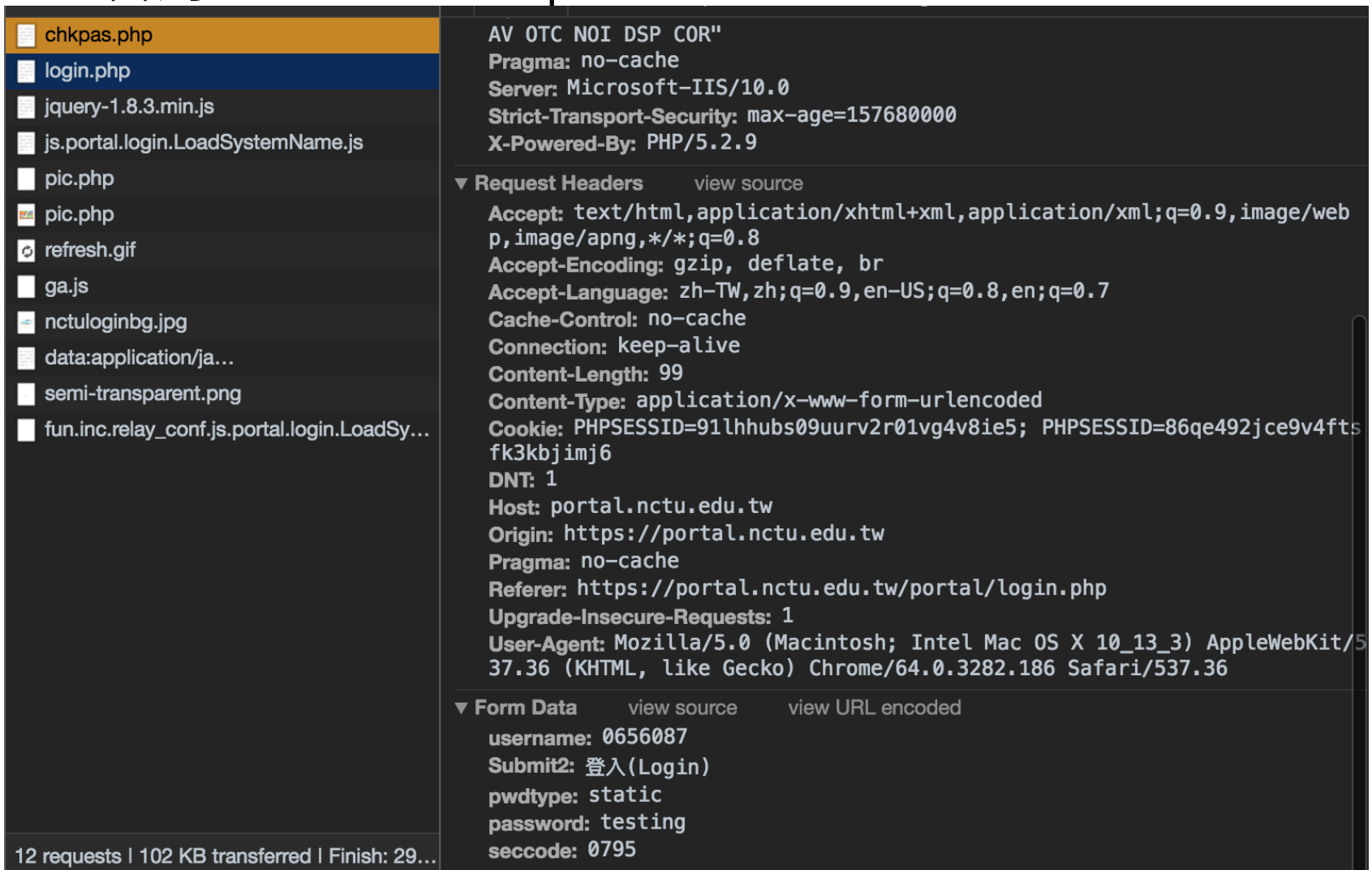
```
➜ 2018NA-homework-1 git:(master) ✗ python3 main.py 0656087
Portal Password:
+------+-------------+--------------------+--------------------+------+--------------------+------+--------------+------+
| 節次 | 時間\星期   | (一)               | (二)               | (三) | (四)               | (五) | (六)         | (日) |
+------+-------------+--------------------+--------------------+------+--------------------+------+--------------+------+
| M    | 06:00~~06:50 |                    |                    |      |                    |      |              |      |
| N    | 07:00~~07:50 |                    |                    |      |                    |      | 個別研究EC015 |      |
| A    | 08:00~~08:50 |                    |                    |      |                    |      |              |      |
| B    | 09:00~~09:50 |                    |                    |      |                    |      |              |      |
| C    | 10:10~~11:00 |                    | 網路安全(英文授課)EC115 |      |                    |      |              |      |
| D    | 11:10~~12:00 |                    | 網路安全(英文授課)EC115 |      |                    |      |              |      |
| X    | 12:20~~13:10 | 數位遊戲與學習ED117 | 網路安全(英文授課)EC115 |      |                    |      |              |      |
| E    | 13:20~~14:10 | 數位遊戲與學習ED117 |                    |      | 軟體測試CS105      |      |              |      |
| F    | 14:20~~15:10 | 數位遊戲與學習ED117 |                    |      | 軟體測試CS105      |      |              |      |
| G    | 15:30~~16:20 |                    |                    |      | 軟體測試CS105      |      |              |      |
| H    | 16:30~~17:20 |                    |                    |      |                    |      |              |      |
| Y    | 17:30~~18:20 |                    |                    |      |                    |      |              |      |
| I    | 18:30~~19:20 |                    |                    |      | 網路管理實務EC122  |      |              |      |
| J    | 19:30~~20:20 |                    |                    |      | 網路管理實務EC122  |      |              |      |
| K    | 20:30~~21:20 |                    |                    |      | 網路管理實務EC122  |      |              |      |
| L    | 21:30~~22:20 |                    |                    |      |                    |      |              |      |
+------+-------------+--------------------+--------------------+------+--------------------+------+--------------+------+
```

1-1: Web Crawler – Hint: Argument Parser Sample

```
➜  2018NA-homework-1 git:(master) ✗ python3 main.py -h
usage: main.py [-h] username

Web crawler for NCTU class schedule.

positional arguments:
  username    username of NCTU portal

optional arguments:
  -h, --help  show this help message and exit
```

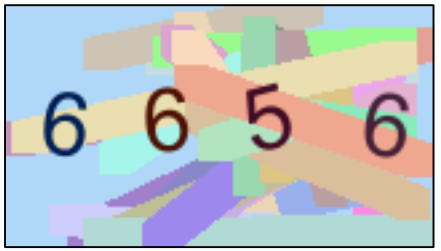
1-1: Web Crawler – Hint

- 善用Browser Development tool → Network



1-1: Web Crawler – Hint: Captcha Recognize

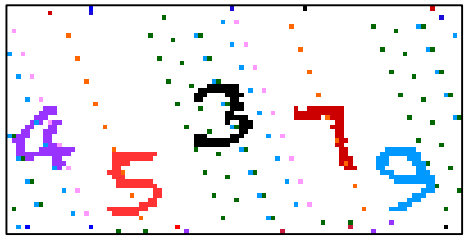
Login Option (a): NCTU Portal



- Preprocess
  - Convert image to grayscale
  - Adjust brightness and contrast
  - Packages may be used
    - Pillow
- Recognize
  - Pytesseract (Python interface for Tesseract OCR By google)

1-1: Web Crawler – Hint: Captcha Recognize

Login Option (b): Course Registration System



- ❑ Preprocess
  - Convert image to grayscale
  - Remove salt and pepper noise
  - Packages may be used
    - ➢ Pillow
    - ➢ OpenCV
- ❑ Recognize
  - Pytesseract (Python interface for Tesseract OCR By google)

1-2: Auth log parser – Requirements (1/6)

- With following option
  - -h show usage 5%
  - -u sort by user 5%
  - -after filter log after special date 10%
  - -before filter log before special date 10%
  - -n show only the user of most #-th times 5%
  - -t show only the user of attacking equal or more than # times 5%
  - -r sort in reverse order 5%
- Only Python script allowed
  - Call shell script is not allowed
  - Call NodeJS script is not allowed
  - `subprocess` not allowed
  - `os.system` not allowed
- Python 3 only

1-2: Auth log parser – Requirements (2/6)

show help

```
➜  2018NA-homework-1 git:(master) ✗ python3 nahw1-2_0656087.py -h
usage: nahw1-2_0656087.py [-h] [-u] [-after AFTER] [-before BEFORE] [-n N]
                          [-t T] [-r]
                          filename

Auth log parser.

positional arguments:
  filename          Log file path.

optional arguments:
  -h, --help        show this help message and exit
  -u                Summary failed login log and sort log by user .
  -after AFTER      Filter log after date. format YYYY-MM-DD-HH:MM:SS
  -before BEFORE    Filter log before date. format YYYY-MM-DD-HH:MM:SS
  -n N              Show only the user of most N-th times
  -t T              Show only the user of attacking equal or more than T times
  -r                Sort in reverse order
```

1-2: Auth log parser – Requirements (3/6)

default output

```
➜  2018NA-homework-1 git:(master) ✗ python3 nahw1-2_0656087.py auth-sample.log
+----------+-------+
|   user   | count |
+----------+-------+
|  admin   |  104  |
|    pi    |   61  |
| service  |   14  |
|   ubnt   |   7   |
|  ethos   |   2   |
|  squid   |   2   |
| skyhertz |   1   |
| usuario  |   1   |
|   user   |   1   |
|  mother  |   1   |
+----------+-------+
```

1-2: Auth log parser – Requirements (4/6)

with -r output

```
[➜  2018NA-homework-1 git:(master) ✗ python3 nahw1-2_0656087.py auth-sample.log -r
+----------+-------+
|   user   | count |
+----------+-------+
| skyhertz |   1   |
| usuario  |   1   |
|   user   |   1   |
|  mother  |   1   |
|  ethos   |   2   |
|  squid   |   2   |
|   ubnt   |   7   |
| service  |   14  |
|    pi    |   61  |
|  admin   |  104  |
+----------+-------+
```

1-2: Auth log parser – Requirements (5/6)

with -u output

```
➜ 2018NA-homework-1 git:(master) ✗ python3 nahw1-2_0656087.py auth-sample.log -u
+----------+-------+
|   user   | count |
+----------+-------+
|  admin   |  104  |
|  ethos   |   2   |
|  mother  |   1   |
|    pi    |   61  |
| service  |   14  |
| skyhertz |   1   |
|  squid   |   2   |
|   ubnt   |   7   |
|   user   |   1   |
| usuario  |   1   |
+----------+-------+
```

1-2: Auth log parser – Requirements (6/6)

with -after -before output

```
➜  2018NA-homework-1 git:(master) ✗ python3 nahw1-2_0656087.py auth-sample.log
 -after 2018-03-08-00:00:00 -before 2018-03-08-10:32:00
+----------+-------+
|   user   | count |
+----------+-------+
|  admin   |   70  |
|    pi    |   40  |
| service  |   7   |
|   ubnt   |   7   |
|  ethos   |   2   |
|  squid   |   2   |
| skyhertz |   1   |
| usuario  |   1   |
|   user   |   1   |
+----------+-------+
```

1-2: Auth log parser – Hint: log sample

- Log format (a)
  - Mar 8 00:48:38 <auth.info> bsd4 sshd[85194]: Invalid user Chiangmj840306 from 36.230.109.223
- Log format (b) without <{facility}.{severity level}>
  - Mar 8 09:50:21 linux7 sshd[15899]: Invalid user ubnt from 188.187.121.68 port 47664
- Set log message year as 2018, if year is not defined on log message.
- No matter how many times the password is tried, each connection is only counted once.
  - Hint: just counting `Invalid user…`
- There is different that include facility and severity level of log message or not between them.
- Your script must can parse both (a) and (b).
- You can get raw sample on https://goo.gl/siimub

Help

- Email to ta@nasa.cs.nctu.edu.tw
- New E3 https://e3new.nctu.edu.tw