Тестирование скорости интернета через подключение к Tor

Проект подготовил ученик 10А класса
Смирнов Павел

Цель

Создать скрипт на языке Python для тестирования скорости интернета через подключение к Tor

Задачи

- Поднять Tor Hidden Service сервер
- Написать скрипт для запросов к Tor Hidden Service через Tor
- Сохранить данные по скорости в CSV таблицу

Актуальность

- Проект актуален для организация, что работают с Tor или для простых людей

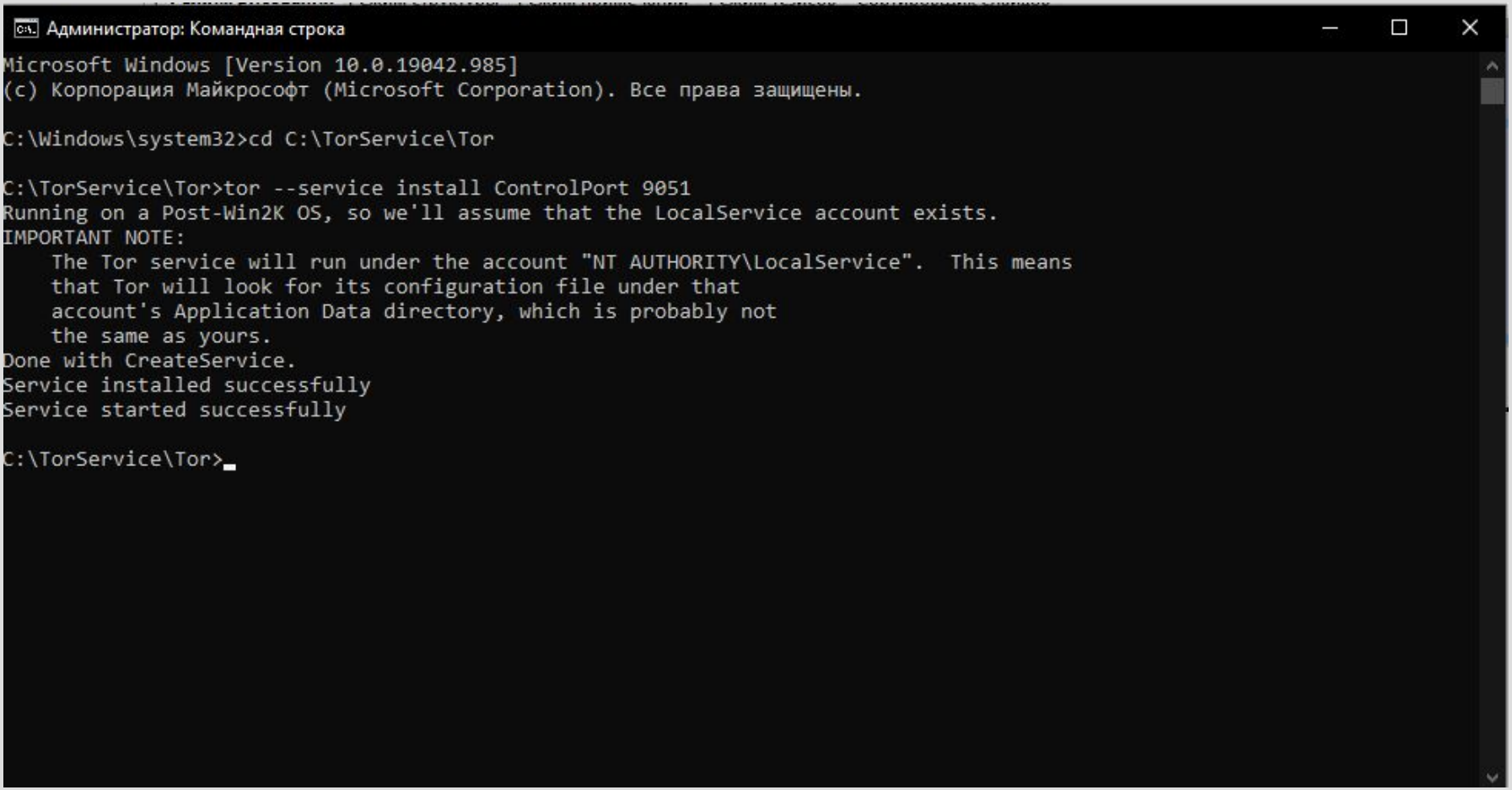
Немного вводной информации

Что такое Tor

Немного вводной информации

Скорость интернета

Установка Tor



Написание Tor Hidden Service сервер

```python
@app.route('/get')
def get():
    return f'{getsize("media/image.gif")} {file_info}'


@app.route('/post')
def post():
    return flask.request.data
```

```python
print(" * Getting controller")
controller = Controller.from_port(address="127.0.0.1", port=int(config['Tor']['ControlPort']))
try:
    controller.authenticate(password=config['Tor']['ServerPassword'])
    controller.set_options([('HiddenServiceDir', '/hidden_service_dir'), ('HiddenServicePort', f'80 {host}:{port}')])
    svc_name = open('/hidden_service_dir' + "/hostname", "r").read().strip()
    print(" * Created host: %s" % svc_name)
    config.set('Tor', 'Url', svc_name)
    with open('config.ini', 'w') as config_file:
```

Отправка запросов на сервер

```
if __name__ == '__main__':
    multiprocessing.freeze_support()
    try:
        print('Tor Requests')
        time_list = config['Schedule']['time'].split()
        for time in time_list:
            if len(time.split(':')[0]) == 1:
                time = '0' + time
            schedule.every().day.at(time).do(test, time)
            print('Set test on', time)
        while True:
            schedule.run_pending()
            sleep(1)
    except Exception as e:
        print(e)
```

Настройка программы через config.ini

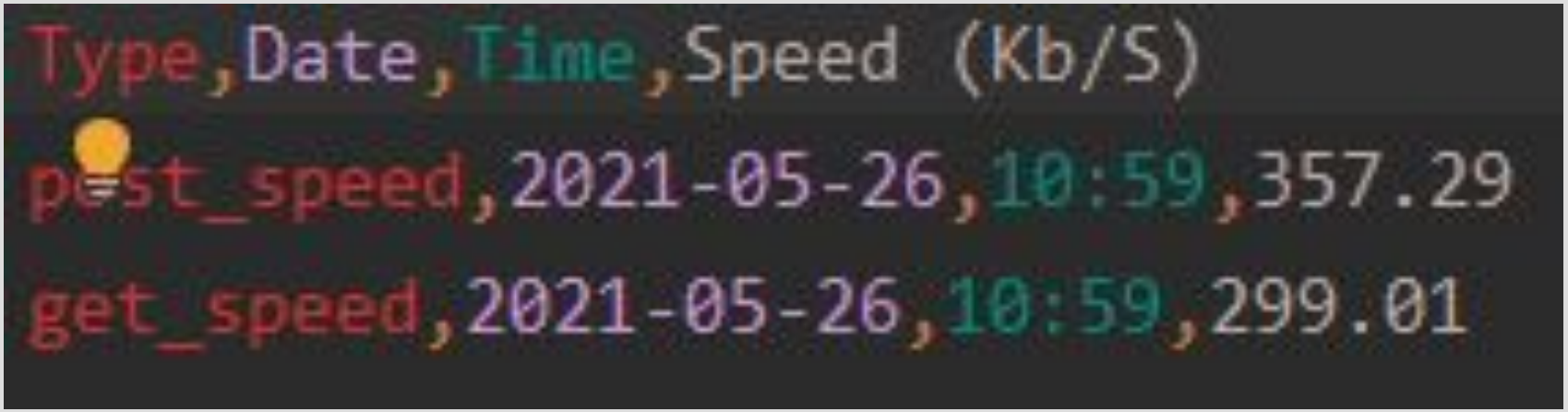
```
[Tor]
controlport = 9051
serverport = 9050
serverpassword =

[Server]
port = 5000

[Schedule]
time =

[Traffic]
min = 10
max = 50
```

Запуск

```
C:\Users\admin\PycharmProjects\TorConnectionSpeedTest\venv\Scripts\python.exe C:/Users/admin/PycharmProjects/TorConnectionSpeedTest/server.py
 * Getting controller
 * Created host: gpeapkutiyudgdta5vpbv4sdmcdc5rps5vuzwkjae3gghsy3vh76pkqd.onion
 * Serving Flask app 'Tor Speed Test Server' (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [26/May/2021 10:52:05] "GET /get HTTP/1.1" 200 -
127.0.0.1 - - [26/May/2021 10:52:23] "GET /post HTTP/1.1" 200 -
```

```
Tor Requests
Set test on 10:52
GET REQUESTS: Объем трафика для теста: 46.0Mb
POST REQUESTS: Объем трафика для теста: 46.0Mb
POST REQUESTS: Завершено: 1 из 40
GET REQUESTS: Завершено: 1 из 40
```

Сохранение информации с csv таблице



Выводы

- Я написал утилиту для тестирования скорости подключения через Tor. Тем самым я познакомился поближе с работой Tor. Все задуманные мною задачи реализованы

Используемые библиотеки

- stem
- requests
- schedule
- flask