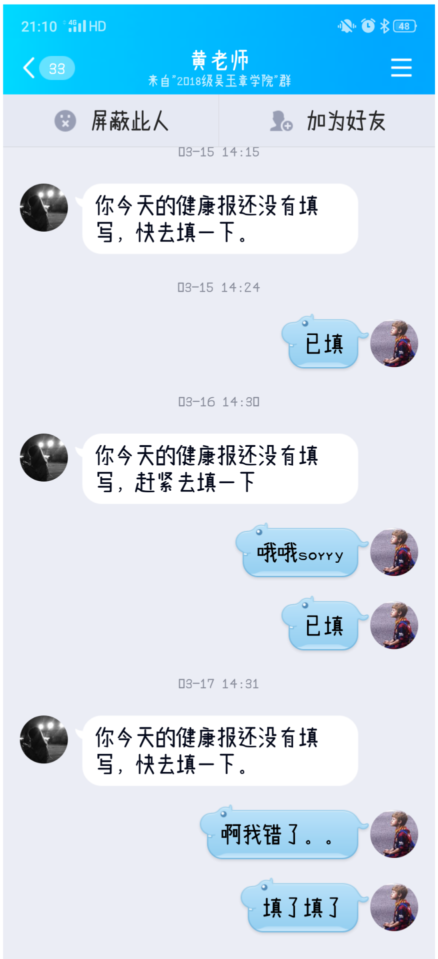
计算机网络实践实验

之前用selenium实现了一个模拟浏览器的操作，但是用浏览器模拟，里面的地点选择api很不稳定（用Firefox的Gecko内核还好一点，Chrome完全就是天马行空，后来就弃用了这个方法，而在这一段时间里面。。。



21:10 HD

黄老师

来自"2018级吴玉章学院"群

03-15 14:15

你今天的健康报还没有填写，快去填一下。

03-15 14:24

已填

03-16 14:30

你今天的健康报还没有填写，赶紧去填一下

哦哦sorry

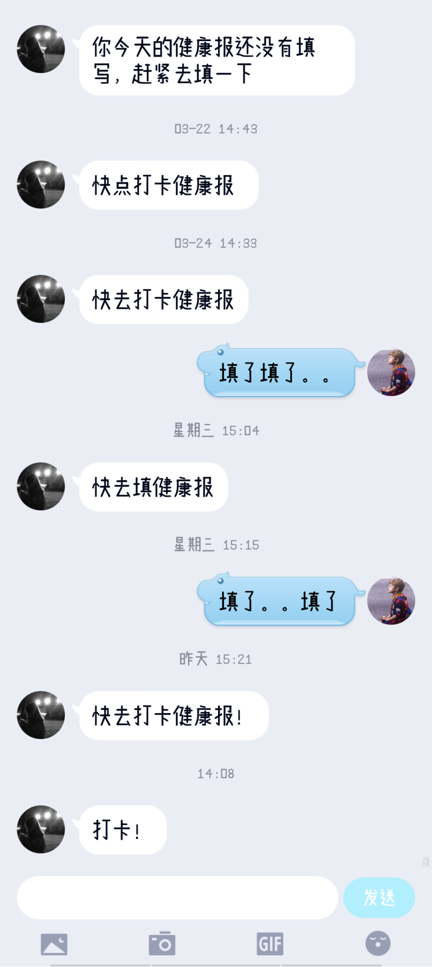
已填

03-17 14:31

你今天的健康报还没有填写，快去填一下。

阿我错了。。

填了填了



你今天的健康报还没有填写，赶紧去填一下

03-22 14:43

快点打卡健康报

03-24 14:33

快去打卡健康报

填了填了。。

星期三 15:04

快去填健康报

星期三 15:15

填了。。填了

昨天 15:21

快去打卡健康报！

14:08

打卡！

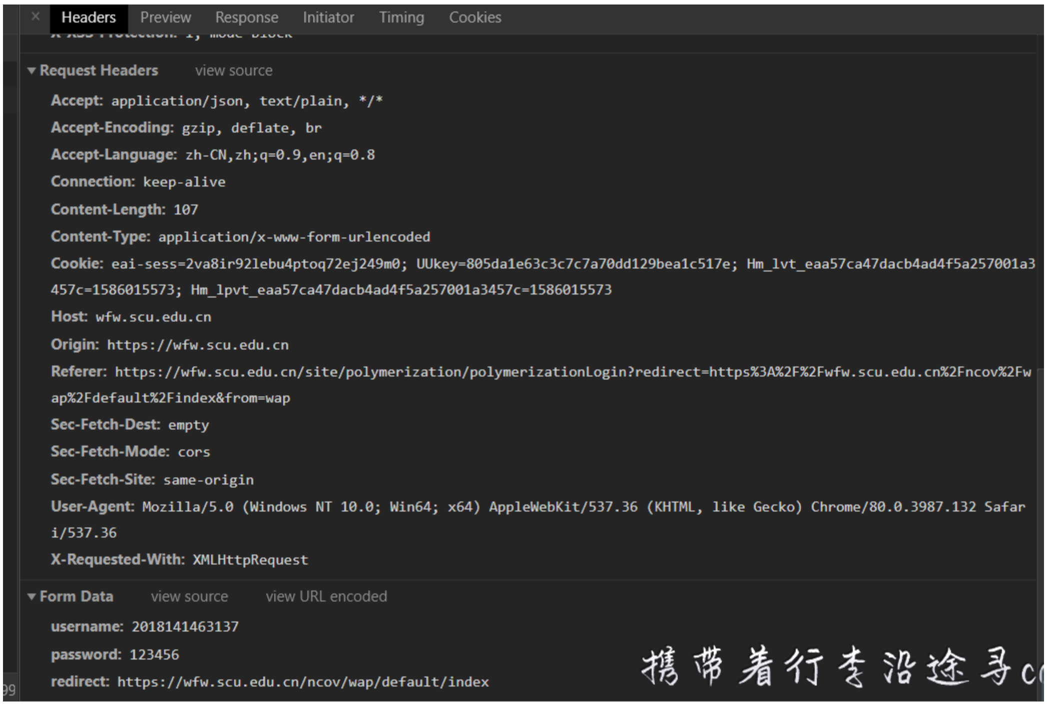
态度逐渐狂躁。。

感到无比愧疚的我，终于从自己的拖延症里自我拉扯出来，抽了点空，做了一下这个计网实验题

思路其实很简单，其实比selenium更好实现，就是模拟post，然后拿到一个eai-sess（Cookie），接着拿这个cookie去登陆，把之前提交过的post再提交一次就是

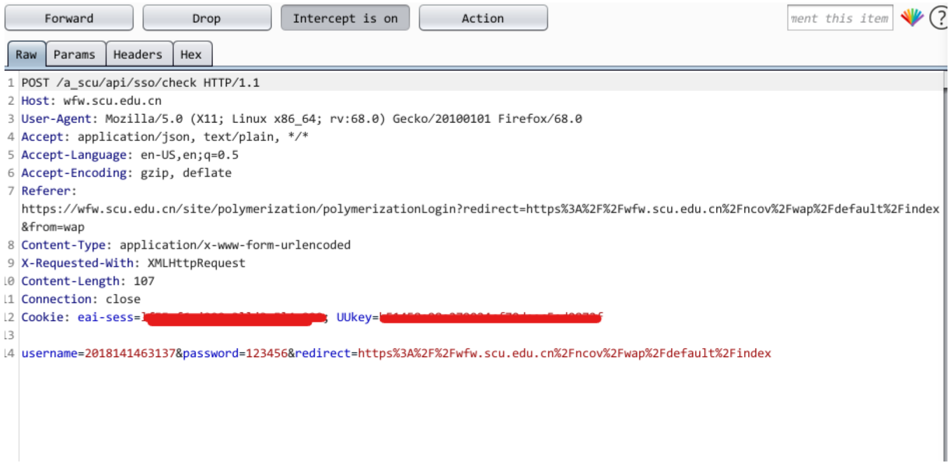
首先先从浏览器打开小程序，得到打卡签到的url:
`https://wfw.scu.edu.cn/ncov/wap/default/index`

进入登陆界面，我们来看看提交的post表单内容。这个抓包分析，可以用自己喜欢的抓包工具，为了不失一般性，我用的Chrome的开发者工具:



```
▾ Request Headers     view source
  Accept: application/json, text/plain, */*
  Accept-Encoding: gzip, deflate, br
  Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
  Connection: keep-alive
  Content-Length: 107
  Content-Type: application/x-www-form-urlencoded
  Cookie: eai-sess=2va8ir92lebu4ptoq72ej249m0; UUkey=805da1e63c3c7c7a70dd129bea1c517e; Hm_lvt_eaa57ca47dacb4ad4f5a257001a3
  457c=1586015573; Hm_lpvt_eaa57ca47dacb4ad4f5a257001a3457c=1586015573
  Host: wfw.scu.edu.cn
  Origin: https://wfw.scu.edu.cn
  Referer: https://wfw.scu.edu.cn/site/polymerization/polymerizationLogin?redirect=https%3A%2F%2Fwfw.scu.edu.cn%2Fncov%2Fw
  ap%2Fdefault%2Findex&from=wap
  Sec-Fetch-Dest: empty
  Sec-Fetch-Mode: cors
  Sec-Fetch-Site: same-origin
  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.132 Safar
  i/537.36
  X-Requested-With: XMLHttpRequest
▾ Form Data     view source     view URL encoded
  username: 2018141463137
  password: 123456
  redirect: https://wfw.scu.edu.cn/ncov/wap/default/index
```

burpsuite或fiddler、wireshark会更强大、直观一点:



```
POST /a_scu/api/sso/check HTTP/1.1
Host: wfw.scu.edu.cn
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
https://wfw.scu.edu.cn/site/polymerization/polymerizationLogin?redirect=https%3A%2F%2Fwfw.scu.edu.cn%2Fncov%2Fwap%2Fdefault%2Findex
&from=wap
Content-Type: application/x-www-form-urlencoded
X-Requested-With: XMLHttpRequest
Content-Length: 107
Connection: close
Cookie: eai-sess=[redacted]; UUkey=[redacted]

username=2018141463137&password=123456&redirect=https%3A%2F%2Fwfw.scu.edu.cn%2Fncov%2Fwap%2Fdefault%2Findex
```

这里有一个有意思的细节，这个request里面存在sessionID。。这是因为和友站共用cookie，下面说的是个人的看法，有错误希望指正

应该是学校的网站没有www主机名，这样，在主机名设置下的cookie会发送到所有的子域，很多的第三方服务网站都是这样共享cookie的，每访问一次网页服务器，浏览器都会发送一次HTTP 和cookie。因此，如果你的cookie设在根域名 “example.com” 下，那么每当你访问 “email.example.com” 或者是 “intranet.example.com” 的时候，浏览器都会发送cookie。

所以，如果你的主机名是根域名(“example.com”)，并且可以登录到内容管理系统(CMS)，那在你登录期间，CMS会给你的浏览器发送一个cookie。接着，如果你访问 “someinternalservice.example.com” (内部服务)，该网站的管理员就可以访问并利用这个cookie，以你的名义登进CMS里的 “example.com” 。类似的，你访问 “email.example.com” (邮箱)的时候，你的CDN服务商也可以登进你的邮件服务，加载出大量含有 “example.com” 的网站，比如 “static.example.com” 等等。

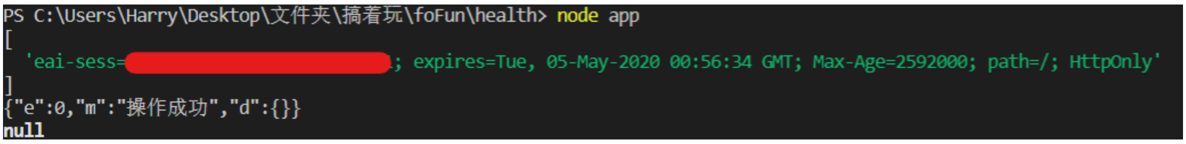
而这样的设计是可能会被利用的，就像ASP.NET的 VIEWSTATE一样，有异曲同工之妙

扯远了，回到这里，我们可以获得url，表单提交方式，已经一些其他参数，我们就可以拿到这些信息去拿cookie了

这里使用的是NodeJs，python和Java都可以比较方便的实现这一部分

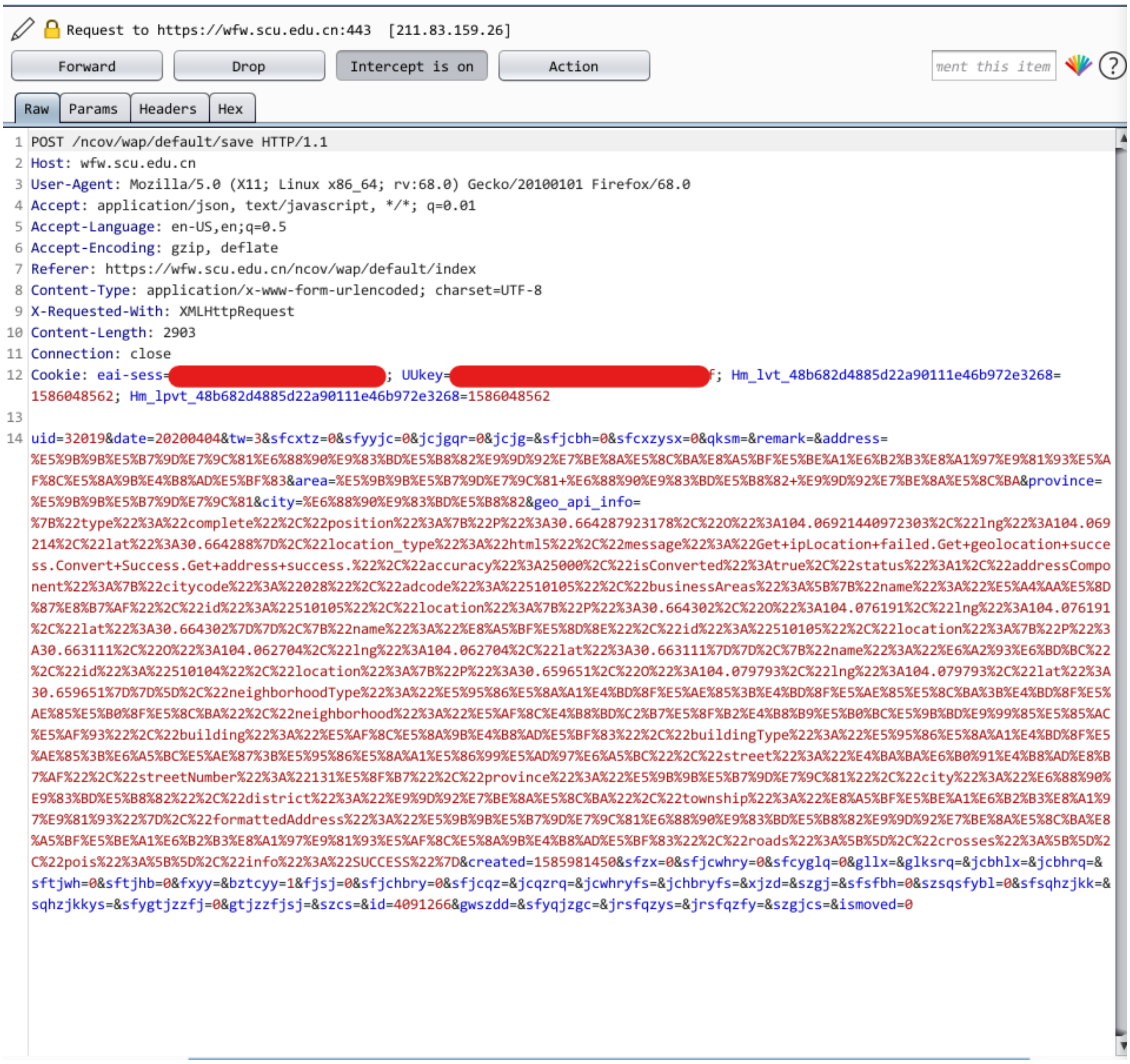
```
let request = require('request');
//登陆post地址
let url = 'https://wfw.scu.edu.cn/a_scu/api/sso/check';

let user = {
    username: '20181414631**',
    password: '******'
};


//设置头部
let headers = {
    'User-Agent': `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36`,
};

let opts = {
    url: url,
    method: 'POST',
    headers: headers,
    form: user,
};
request(opts,(err,res,body)=>{
    let cookie=res.['set-cookie']
    console.log(cookie)
})
```

打印出来cookie，没什么问题



```
PS C:\Users\Harry\Desktop\文件夹\搞着玩\foFun\health> node app
[
  'eai-sess=[redacted]; expires=Tue, 05-May-2020 00:56:34 GMT; Max-Age=2592000; path=/; HttpOnly'
]
{"e":0,"m":"操作成功","d":{}}
null
```

登陆之后我们抓一下最后提交的包



```
POST /ncov/wap/default/save HTTP/1.1
Host: wfw.scu.edu.cn
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://wfw.scu.edu.cn/ncov/wap/default/index
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 2903
Connection: close
Cookie: eai-sess=[redacted]; UUkey=[redacted]f; Hm_lvt_48b682d4885d22a90111e46b972e3268=
1586048562; Hm_lpvt_48b682d4885d22a90111e46b972e3268=1586048562

uid=32019&date=20200404&tw=3&sfcxtz=0&sfyyjc=0&jcjgqr=0&jcjg=&sfjcbh=0&sfcxzysx=0&qksm=&remark=&address=
%E5%9B%9B%E5%B7%9D%E7%9C%81%E6%88%90%E9%83%BD%E5%B8%82%E9%9D%92%E7%BE%8A%E5%8C%BA%E8%A5%BF%E5%BE%A1%E6%B2%B3%E8%A1%97%E9%81%93%E5%A
F%8C%E5%8A%9B%E4%B8%AD%E5%BF%83&area=%E5%9B%9B%E5%B7%9D%E7%9C%81+%E6%88%90%E9%83%BD%E5%B8%82+%E9%9D%92%E7%BE%8A%E5%8C%BA&province=
%E5%9B%9B%E5%B7%9D%E7%9C%81&city=%E6%88%90%E9%83%BD%E5%B8%82&geo_api_info=
%7B%22type%22%3A%22complete%22%2C%22position%22%3A%7B%22P%22%3A30.664287923178%2C%22O%22%3A104.06921440972303%2C%22lng%22%3A104.069
214%2C%22lat%22%3A30.664288%7D%2C%22location_type%22%3A%22html5%22%2C%22message%22%3A%22Get+ipLocation+failed.Get+geolocation+succe
ss.Convert+Success.Get+address+success.%22%2C%22accuracy%22%3A25000%2C%22isConverted%22%3Atrue%2C%22status%22%3A1%2C%22addressCompo
nent%22%3A%7B%22citycode%22%3A%22028%22%2C%22adcode%22%3A%22510105%22%2C%22businessAreas%22%3A%5B%7B%22name%22%3A%22%E5%A4%AA%E5%8D
%87%E8%B7%AF%22%2C%22id%22%3A%22510105%22%2C%22location%22%3A%7B%22P%22%3A30.664302%2C%22O%22%3A104.076191%2C%22lng%22%3A104.076191
%2C%22lat%22%3A30.664302%7D%7D%2C%7B%22name%22%3A%22%E8%A5%BF%E5%8D%8E%22%2C%22id%22%3A%22510105%22%2C%22location%22%3A%7B%22P%22%3
A30.663111%2C%22O%22%3A104.062704%2C%22lng%22%3A104.062704%2C%22lat%22%3A30.663111%7D%7D%2C%7B%22name%22%3A%22%E6%A2%93%E6%BD%BC%22
%2C%22id%22%3A%22510104%22%2C%22location%22%3A%7B%22P%22%3A30.659651%2C%22O%22%3A104.079793%2C%22lng%22%3A104.079793%2C%22lat%22%3A
30.659651%7D%7D%5D%2C%22neighborhoodType%22%3A%22%E5%95%86%E5%8A%A1%E4%BD%8F%E5%AE%85%3B%E4%BD%8F%E5%AE%85%E5%8C%BA%3B%E4%BD%8F%E5%
AE%85%E5%B0%8F%E5%8C%BA%22%2C%22neighborhood%22%3A%22%E5%AF%8C%E4%B8%BD%C2%B7%E5%8F%B2%E4%B8%B9%E5%B0%BC%E5%9B%BD%E9%99%85%E5%85%AC
%E5%AF%93%22%2C%22building%22%3A%22%E5%AF%8C%E5%8A%9B%E4%B8%AD%E5%BF%83%22%2C%22buildingType%22%3A%22%E5%95%86%E5%8A%A1%E4%BD%8F%E5
%AE%85%3B%E6%A5%BC%E5%AE%87%3B%E5%95%86%E5%8A%A1%E5%86%99%E5%AD%97%E6%A5%BC%22%2C%22street%22%3A%22%E4%BA%BA%E6%B0%91%E4%B8%AD%E8%B
7%AF%22%2C%22streetNumber%22%3A%22131%E5%8F%B7%22%2C%22province%22%3A%22%E5%9B%9B%E5%B7%9D%E7%9C%81%22%2C%22city%22%3A%22%E6%88%90%
E9%83%BD%E5%B8%82%22%2C%22district%22%3A%22%E9%9D%92%E7%BE%8A%E5%8C%BA%22%2C%22township%22%3A%22%E8%A5%BF%E5%BE%A1%E6%B2%B3%E8%A1%9
7%E9%81%93%22%7D%2C%22formattedAddress%22%3A%22%E5%9B%9B%E5%B7%9D%E7%9C%81%E6%88%90%E9%83%BD%E5%B8%82%E9%9D%92%E7%BE%8A%E5%8C%BA%E8
%A5%BF%E5%BE%A1%E6%B2%B3%E8%A1%97%E9%81%93%E5%AF%8C%E5%8A%9B%E4%B8%AD%E5%BF%83%22%2C%22roads%22%3A%5B%5D%2C%22crosses%22%3A%5B%5D%2
C%22pois%22%3A%5B%5D%2C%22info%22%3A%22SUCCESS%22%7D&created=1585981450&sfzx=0&sfjcwhry=0&sfcyglq=0&gllx=&glksrq=&jcbhlx=&jcbhrq=&
sftjwh=0&sftjhb=0&fxyy=&bztcyy=1&fjsj=0&sfjchbry=0&sfjcqz=&jcqzrq=&jcwhryfs=&jchbryfs=&xjzd=&szgj=&sfsfbh=0&szsqsfybl=0&sfsqhzjkk=&
sqhzjkkys=&sfygtjzzfj=0&gtjzzfjsj=&szcs=&id=4091266&gwszdd=&sfyqjzgc=&jrsfqzys=&jrsfqzfy=&szgjcs=&ismoved=0
```

将这个包里面的内容用刚刚的cookie提交即可

即cookie:res.headers['set-cookie'],还有一个要注意的,就是date是昨天。。。虽然不知道为什么这样

最后加上时间监听就好啦

```
var request = require('request');
var time =require('china-time')
var schedule = require('node-schedule');

//登陆post地址
let url = 'https://wfw.scu.edu.cn/a_scu/api/sso/check';

//设置头部
let headers = {
    'User-Agent': `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36`,
};
//post 内容
let opts = {
    url: url,
    method: 'POST',
    headers: headers,
    form:'username=2018141463137&password=******'
```

```
};

//模拟登陆



const scheduleCronstyle = () => {
    schedule.scheduleJob('0 0 12 * * *', () => {
        console.log(time('YYYYMMDD'))
        request(opts, (err, res, body) => {
            let opts = {
                url: 'https://wfw.scu.edu.cn/ncov/wap/default/save',
                headers: {
                    'User-Agent': `Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36`,
                    Cookie: res.headers['set-cookie'],
                },
                form:
`uid=32019&date=${time('YYYYMMDD')-1}&tw=3&sfcxtz=0&sfyyjc=0&jcjgqr=0&jcjg=&sfjc
bh=0&sfcxzysx=0&qksm=&remark=&address=%E5%9B%9B%E5%B7%9D%E7%9C%81%E6%88%90%E9%83
%BD%E5%B8%82%E9%9D%92%E7%BE%8A%E5%8C%BA%E8%A5%BF%E5%BE%A1%E6%B2%B3%E8%A1%97%E9%8
1%93%E5%AF%8C%E5%8A%9B%E4%B8%AD%E5%BF%83&area=%E5%9B%9B%E5%B7%9D%E7%9C%81+%E6%88
%90%E9%83%BD%E5%B8%82+%E9%9D%92%E7%BE%8A%E5%8C%BA&province=%E5%9B%9B%E5%B7%9D%E7
%9C%81&city=%E6%88%90%E9%83%BD%E5%B8%82&geo_api_info=%7B%22type%22%3A%22complete
%22%2C%22position%22%3A%7B%22P%22%3A30.66428792317 8%2C%22O%22%3A104.069214409723
03%2C%22lng%22%3A104.069214%2C%22lat%22%3A30.664288%7D%2C%22location_type%22%3A%
22html5%22%2C%22message%22%3A%22Get+ipLocation+failed.Get+geolocation+success.Co
nvert+Success.Get+address+success.%22%2C%22accuracy%22%3A25000%2C%22isConverted%
22%3Atrue%2C%22status%22%3A1%2C%22addressComponent%22%3A%7B%22citycode%22%3A%220
28%22%2C%22adcode%22%3A%22510105%22%2C%22businessAreas%22%3A%5B%7B%22name%22%3A%
22%E5%A4%AA%E5%8D%87%E8%B7%AF%22%2C%22id%22%3A%22510105%22%2C%22location%22%3A%7
B%22P%22%3A30.664302%2C%22O%22%3A104.076191%2C%22lng%22%3A104.076191%2C%22lat%22
%3A30.664302%7D%7D%2C%7B%22name%22%3A%22%E8%A5%BF%E5%8D%8E%22%2C%22id%22%3A%2251
0105%22%2C%22location%22%3A%7B%22P%22%3A30.663111%2C%22O%22%3A104.062704%2C%22ln
g%22%3A104.062704%2C%22lat%22%3A30.663111%7D%7D%2C%7B%22name%22%3A%22%E6%A2%93%E
6%BD%BC%22%2C%22id%22%3A%22510104%22%2C%22location%22%3A%7B%22P%22%3A30.659651%2
C%22O%22%3A104.079793%2C%22lng%22%3A104.079793%2C%22lat%22%3A30.659651%7D%7D%5D%
2C%22neighborhoodType%22%3A%22%E5%95%86%E5%8A%A1%E4%BD%8F%E5%AE%85%3B%E4%BD%8F%E
5%AE%85%E5%8C%BA%3B%E4%BD%8F%E5%AE%85%E5%B0%8F%E5%8C%BA%22%2C%22neighborhood%22%
3A%22%E5%AF%8C%E4%B8%BD%C2%B7%E5%8F%B2%E4%B8%B9%E5%B0%BC%E5%9B%BD%E9%99%85%E5%85
%AC%E5%AF%93%22%2C%22building%22%3A%22%E5%AF%8C%E5%8A%9B%E4%B8%AD%E5%BF%83%22%2C
%22buildingType%22%3A%22%E5%95%86%E5%8A%A1%E4%BD%8F%E5%AE%85%3B%E6%A5%BC%E5%AE%8
7%3B%E5%95%86%E5%8A%A1%E5%86%99%E5%AD%97%E6%A5%BC%22%2C%22street%22%3A%22%E4%BA%
BA%E6%B0%91%E4%B8%AD%E8%B7%AF%22%2C%22streetNumber%22%3A%22131%E5%8F%B7%22%2C%22
province%22%3A%22%E5%9B%9B%E5%B7%9D%E7%9C%81%22%2C%22city%22%3A%22%E6%88%90%E9%8
3%BD%E5%B8%82%22%2C%22district%22%3A%22%E9%9D%92%E7%BE%8A%E5%8C%BA%22%2C%22towns
hip%22%3A%22%E8%A5%BF%E5%BE%A1%E6%B2%B3%E8%A1%97%E9%81%93%22%7D%2C%22formattedAd
dress%22%3A%22%E5%9B%9B%E5%B7%9D%E7%9C%81%E6%88%90%E9%83%BD%E5%B8%82%E9%9D%92%E7
%BE%8A%E5%8C%BA%E8%A5%BF%E5%BE%A1%E6%B2%B3%E8%A1%97%E9%81%93%E5%AF%8C%E5%8A%9B%E
4%B8%AD%E5%BF%83%22%2C%22roads%22%3A%5B%5D%2C%22crosses%22%3A%5B%5D%2C%22pois%22
%3A%5B%5D%2C%22info%22%3A%22SUCCESS%22%7D&created=1585981450&sfzx=0&sfjcwhry=0&s
fcyglq=0&gllx=&glksrq=&jcbhlx=&jcbhrq=&sftjwh=0&sftjhb=0&fxyy=&bztcyy=1&fjsj=0&s
fjchbry=0&sfjcqz=&jcqzrq=&jcwhryfs=&jchbryfs=&xjzd=&szgj=&sfsfbh=0&szsqsfybl=0&s
fsqhzjkk=&sqhzjkkys=&sfygtjzzfj=0&gtjzzfjsj=&szcs=&id=4091266&gwszdd=&sfyqjzgc=&
jrsfqzys=&jrsfqzfy=&szgjcs=&ismoved=0`

            }
            request(opts, (err, res, body) => {
                console.log(body)
            })
        });

    });
}
```