某数通用爬虫

以下内容仅供内部学习交流使用，请勿用于商业及违法用途，后果自负！！

先分享之前搜集的大佬的瑞数文章（还有很多，忘记保存了）：

Python 爬虫进阶必备 | R数 4 代加密分析https://mp.weixin.qq.com/s/lFUSFxokzJaWjZ3NGLJs4Q

【瑞数】维普期刊JS逆向4000字详细流程_1_获取接口签名（看到了十一姐😄）
https://blog.csdn.net/qq_35491275/article/details/117307069?spm=1001.2014.3001.5502

JS逆向 | 瑞数四代补环境获取$_ts和eval
https://mp.weixin.qq.com/s/Z2VfT1sQi9NSZYIsAym09Q

【Javascript】基于Jsdom补环境(瑞数进阶篇一)
http://120.24.42.85/index.php/archives/72/

js逆向瑞数5解密讲解与插桩讲解（非常详细，建议看看）
https://www.bilibili.com/video/BV1PS4y127Mn

好了，假设你已经看过上面的文章或者早就是瑞数大佬了

下面分享一下使用playwright这个框架来爬取fangdi

首先，瑞数hook了xhr，在rs的网站一般都是下面这样（高版本略有不同）

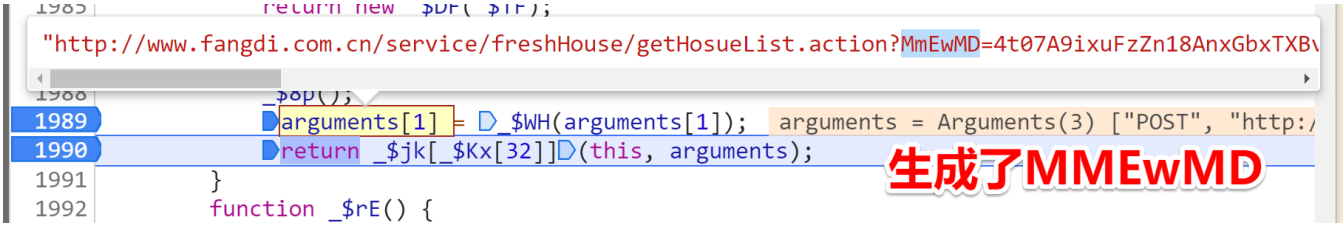
```
> XMLHttpRequest.prototype.open
ƒ _$zw(){_$8p();arguments[1]=_$WH(arguments[1]);return _$jk[_$Kx[32]](this,arguments);}
>
```

在这个函数生成了url中的参数MmEwMD

```
986         }
987         function _$zw() {
988             _$8p();
989             arguments[1] = _$WH(arguments[1]);
990             return _$jk[_$Kx[32]](this, arguments);
991         }
992         function _$rE() {
993             return _$At(new _$Eo(), false);
```

"/service/freshHouse/getHosueList.action"

传进来的url



好了，看到这里本文已经完结，用RPC无限生成就搞定了（狗头）

当然还有cookie生成了，cookie生成其实很简单，rs的网站只要点击一下鼠标就会自动产生新cookie，利用鼠标点击事件就OK，下面上代码了（非常简单）

```
from playwright.sync_api import sync_playwright

"""
rs playwright
"""


playwright = sync_playwright().start()


# 防止webdriver检测
hook_webdriver = """
Object.defineProperty(navigator, 'webdriver', {
      get: () => undefined
    })
"""
# 关闭自动化特征
arg = ['--disable-blink-features=AutomationControlled']
browser = playwright.chromium.launch(headless=False, slow_mo=100, args=arg)
# storage_state加载 cookie，跳过登录，大部分不需要登录
# context = browser.new_context(storage_state='fangdi.json')
context = browser.new_context()


# 这里可以提前加载js，类似于浏览器插件，这里加载过webdriver检测的js
context.add_init_script(hook_webdriver)
# context.add_init_script(hook_js)


# 新建页面
page = context.new_page()
page.goto('http://www.fangdi.com.cn/new_house/new_house_list.html')
# 保存cookie
context.storage_state(path='fangdi.json')
# # 获取window对象
# window = page.evaluate_handle("window")
# window.evaluate('navigator.userAgent')
```

```
# api
url = "http://www.fangdi.com.cn/service/freshHouse/getHosueList.action"

# 请求参数
data = {
    "districtID": "",
    "dicRegionID": "",
    "stateID": "",
    "houseAreaID": "",
    "dicAvgpriceID": "",
    "dicPositionID": "",
    "houseTypeID": "",
    "address": "",
    "openingID": "",
    "projectName": "",
    "currentPage": "1"
}
# 发起post请求，类似于ajax请求，也可以使用fetch
page.mouse.click(1, 1)  # 产生新cookie
res = page.request.post(url, form=data)
# page.request.fetch()
print(res.text())  # json格式 res.json()


data['currentPage'] = 3
print(page.request.post(url, form=data).text())


page.close()
browser.close()
```

代码其实很简单，这里说一下为什么 page.request.post 就可以直接请求，我们并没有调用rpc之类的代码。
request.post请求是浏览器直接发起的，类似于我们点击网站时发起的ajax请求，所以rs对xhr的hook也是生效的，简单来说就是我们的post也被自动加上了MmEwMD参数，
page.mouse.click(1, 1)产生了新的cookie，post请求时会自动加上，cookie很简单的就解决了。

最后
fangdi已经被大佬们分析透了，也就是rs4，这篇文章没有提到rs版本，因为其实是通用的，本身走的就是和浏览器一样的调用方式。但是，没有考虑新版rs的风控，鼠标事件，指纹等等，写太多容易被查水表，就当是一个新的RPC思路吧，毕竟没有使用第三方rpc库！
文章很简洁，顺便安利下playwright，拿来做rpc太好了呀！

保存到我的笔记