

说在前面

在渗透测试及漏洞挖掘过程中，信息搜集是一个非常重要的步骤。而在网站的JS文件中，会存在各种对测试有帮助的内容。

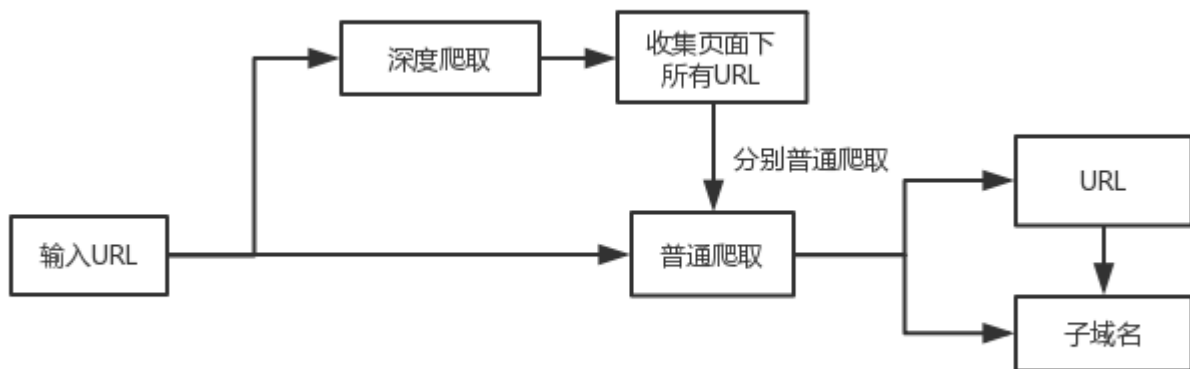
比如：敏感接口，子域名等。

社区内的文章也有有些关于JS文件提取信息的片段，比如Brupsuite和LinkFinder结合的方式，但还是有些问题：不能提取子域名，是相对URL，没那么方便等等。

于是我写了一个工具 - JSFinder。能够根据一个URL自动的收集JS，并在其中发现提取URL和子域名。毕竟，信息搜集的方式，自然是越多越好。

项目地址：<https://github.com/Thre3zh1/JSFinder>

JSFinder获取URL和子域名的方式：



先知社区

使用方式

我们以京东为例来测试，京东的网址为：<https://www.jd.com/>

简单爬取：

```
python3 JSFinder.py -u https://www.jd.com/
```

提取的URL：

```
root@kali:~/Threezh1/Temp# python3 JSFinder.py -u https://www.jd.com/
url:https://www.jd.com/
Find 178 URL:
https://s-nfa.jd.com/bd?info=
https://passport.jd.com/new/login.aspx?ReturnUrl=
https://reg.jd.com/reg/person?ReturnUrl=
https://loadAsyncError.jd.com
https://lapi.jd.com/cookie/check?source=pc-home
https://miaosha.jd.com
https://miaosha.jd.com/pinpailist.html
https://miaosha.jd.com/brandlist.html
https://miaosha.jd.com/#
https://top.jd.com/
https://jdjz.jd.com/
https://jdiscover.jd.com
https://cart.jd.com/
https://huiguang.jd.com
https://hellojoy.jd.com/
https://storage.jd.com/7ff749b346d2f947/3741125c32.js
https://mercury.jd.com/log.gif?t=rec.619066&v=src=rec$errorcode=
https://passport.jd.com/loginservice.aspx?method=Login
https://passport.jd.com/user/petName/getUserInfoForMiniJd.action
https://floor.jd.com/user/score/get
https://pjapi.jd.com/user/scoreAndLevel?source=pc_home
https://passport.jd.com/new/helloService.ashx
https://ai.jd.com/index_new.php?app=Newuser&action=isNewuser
https://dy.jd.com/jsf_user_level
https://corp.jd.com/publicSoa/userInfo/getUserLevel
https://floor.jd.com/user/hotwords/get
https://storage.jd.com/7ff749b346d2f947/2b340c1a7e.js
https://d.jd.com/client/get
https://d.jd.com/navigation/get
https://order.jd.com/lazy/getOrderListCountJson.action
https://question.jd.com/myjd/getMyJdAnswerCount.action
https://pjapi.jd.com/followCommodity/queryForCountByReduceProductAndPin?source=pc_home&sysName=misc
https://quan.jd.com/getcouponcount.action
https://joycenter.jd.com/msgCenter/init.action
https://btshow.jd.com/iou/queryBT.do?sourceType=137
https://diviner.jd.com/diviner?p=610009&lid=1
https://storage.jd.com/7ff749b346d2f947/7a55efa35e.js
https://floor.jd.com/user/feed/get
https://storage.jd.com/fd1f0b5a9b65f5e8/d643e3b2c8.js
https://yuding.jd.com/presaleInfo/getPresaleInfo.action
https://dy.jd.com/jsf_selection_online
https://dy.jd.com/jsf_yy_hide_price
https://pjapi.jd.com/filterSku/preSaleReservation?source=pc_home
https://storage.jd.com/7ff749b346d2f947/143d03c33e.js
https://papi-service.jd.com/feed/content/aggregate/?format=jsonp
https://storage.jd.com/fd1f0b5a9b65f5e8/30a9de95f5.js
https://papi-service.jd.com/feed/content/async/?format=jsonp
https://diviner.jd.com/diviner?p=619028&lid=1&ec=utf-8
https://floor.jd.com/recommend/news/get
https://floor.jd.com/user/score/get?auth=qazwsc
https://portal.cms.jd.com/preview/preview/preview/6153
https://wq.jd.com/webmonitor/collect/badjs.json?Content=
https://chongzhi.jd.com/jdhome-czindex-2017.html
https://floor.jd.com/recommend/lbs/get?t=
https://nfa.jd.com/loadFa.action?aid=0_0_8857
https://nfa.jd.com/loadFa.js?aid=2_955_8766
https://search.jd.com/Search?keyword={keyword}&enc={enc}{additional}
https://search-e.jd.com/searchDigitalBook?ajaxSearch=0&enc=utf-8&key={keyword}&page=1{additional}
https://music.jd.com/8_0_desc_0_0_1_15.html?key={keyword}
https://s-e.jd.com/Search?key={keyword}&enc=utf-8
https://mall.jd.com/index-#{shop_id}.html
https://hiswd.jd.com/?pvid=
https://suggest-squanqiu.jd.com/?terminal=shouquanqiu
https://dd-search.jd.com/?terminal=pc&newjson=1
https://search.jd.com/image?op=upload
https://search.jd.com/image?path=
```

先知社区

先知社区

提取的子域名：

Find 81 Subdomain:
s-nfa.jd.com
passport.jd.com
reg.jd.com
loadAsyncError.jd.com
lapi.jd.com
miaosha.jd.com
top.jd.com
jdzj.jd.com
jdiscover.jd.com
cart.jd.com
huiguang.jd.com
hellojoy.jd.com
kuaibao.jd.com
home.jd.com
xinren.jd.com
vip.jd.com
ypzj.jd.com
fxhh.jd.com
mall.jd.com
haodian.jd.com
plus.jd.com
sale.jd.com
b.jd.com
bvip.jd.com
jdlive.jd.com
nfa.jd.com
storage.jd.com
ch.jd.com
ai.jd.com
dy.jd.com
pjapi.jd.com

打开一个像接口的URL看看

← → ↺  <https://papi-service.jd.com/feed/content/aggregate/?format=jsonp>

[illegible]

看起来是一个商品信息的接口。

只有一百多个URL和几十个子域名，远远不够。

当你想获取更多信息的时候，可以使用-d进行深度爬取来获得更多内容，并使用命令 -ou, -os来指定URL和子域名所保存的文件名。

```
python3 JSFinder.py -u https://www.jd.com/ -d -ou jd_url.txt -os jd_domain.txt
```

```
root@kali:~/Threezh1/Temp# python3 JSFinder.py -u https://www.jd.com/ -d -ou jd_url.txt -os jd_domain.txt
ALL Find 129 links
url:https://www.jd.comjavascript:login();
Fail to access https://www.jd.comjavascript:login();
url:https://www.jd.comjavascript:regist();
Fail to access https://www.jd.comjavascript:regist();
url:https://order.jd.com/center/list.action
Remaining 129 | Find 12 URL in https://order.jd.com/center/list.action
url:https://home.jd.com/
Remaining 128 | Find 44 URL in https://home.jd.com/
url:https://vip.jd.com/
Remaining 127 | Find 71 URL in https://vip.jd.com/
url:https://b.jd.com/
Remaining 126 | Find 312 URL in https://b.jd.com/
url:https://www.jd.com
Remaining 125 | Find 178 URL in https://www.jd.com
url:https://cart.jd.com/cart.action
Remaining 124 | Find 30 URL in https://cart.jd.com/cart.action
url:https://miaosha.jd.com/
Remaining 123 | Find 20 URL in https://miaosha.jd.com/
url:https://a.jd.com/
Remaining 122 | Find 44 URL in https://a.jd.com/
url:https://plus.jd.com/index?flow system=appicon&flow entrance=appicon11&flow channel=pc
gate.jd.com
sale1.jd.com
qiang.jd.com
my.jd.com
huan.jd.com
wan.jd.com
card.jd.com
jiayouka.jd.com
movie.jd.com
mygiftcard.jd.com

Output 4019 urls ←
Path:jd_url.txt

Output 319 subdomains ←
Path:jd_domain.txt
```

4019个URL,319个子域名，能够收集到的内容还是非常多的。

当然，信息的质量取决于网站，各种接口有没有用还取决于自己。

除了这两种方式以外，还可以批量指定URL和JS链接来获取里面的URL。

指定URL：

```
python JSFinder.py -f text.txt
```

指定JS：

```
python JSFinder.py -f text.txt -j
```

最后

写这个脚本的目的是为了丰富信息搜集，也是锻炼自己的编程能力。如果师傅们有更好的建议，希望能够告诉我，谢谢。

email:xiaothreezhi@gmail.com

点击收藏 | 5 关注 | 2

[上一篇：基于污点分析的XSS漏洞辅助挖掘的...](#) [下一篇：【CISCN2019】华北赛区-天...](#)

1. 10 条回复



[lucifer法哥](#) 2019-06-13 14:13:07

很有用

0 回复Ta



[onl****uige](#) 2019-07-26 11:49:32

不错，可以集成到自动化里面。

0 回复Ta



[fayewong菲菲](#) 2019-09-13 10:01:34


```
D:\Python27\JSFinder>JSFinder.py
Traceback (most recent call last):
  File "D:\Python27\JSFinder\JSFinder.py", line 8, in <module>
    from urllib.parse import urlparse
ImportError: No module named parse

D:\Python27\JSFinder>_
```

先知社区

请问出现如图的情况，怎么使用

0 回复Ta



[Threeth1](#) 2019-09-15 15:09:15

[@fayewong菲菲](#) 百度一下，安装这个模块。

0 回复Ta



[fayewong菲菲](#) 2019-09-19 21:08:18

[@Threeth1](#) 好的，谢谢

0 回复Ta



[九守](#) 2019-09-27 12:18:49

js挖洞小火一把，目测此文章要火

0 回复Ta



[Mke2fs](#) 2019-09-27 12:20:29

[@fayewong菲菲](#) 用python3可以直接跑

0 回复Ta



[fayewong菲菲](#) 2019-10-05 10:55:42

@Mke2fs 确定吗，怎么我这里一直提示确实parse模块，哪位大神帮忙看看!

```
D:\Python37\JSFinder>JSFinder.py -u https://www.baidu.com
Traceback (most recent call last):
  File "D:\Python37\JSFinder\JSFinder.py", line 8, in <module>
    from urllib.parse import urlparse
ImportError: No module named parse

D:\Python37\JSFinder>
```

先知社区

0 回复Ta



[fayewong菲菲](#) 2019-10-05 11:00:20

```
D:\Python37\Scripts>pip3 install parse
Requirement already satisfied: parse in d:\python37\lib\site-packages (1.12.1)

D:\Python37\Scripts>
```

先知社区

0 回复Ta



[fayewong菲菲](#) 2019-10-05 11:06:38

装个linkfinder也是同样的情况

```
D:\Python37\linkfinder>linkfinder.py
```

```
Traceback (most recent call last):
```

```
File "D:\Python37\linkfinder\linkfinder.py", line 11, in <module>
```

```
import re, sys, glob, html, argparse, jsbeautifier, webbrowser, subprocess, base64, ssl, xml.etree.ElementTree
```


ImportError: No module named html
目录也加进去了
D:\Python37>python.exe</module>

```
import sys
print(sys.path)
['', 'D:\Python37\python37.zip', 'D:\Python37\DLLs', 'D:\Python37\lib', 'D:\Python37', 'D:\Python37\lib\site-packages']
```

0 回复Ta

[登录](#) 后跟帖

先知社区

[现在登录](#)

热门节点

[技术文章](#)

[社区小黑板](#)

目录

[RSS](#) [关于社区](#) [友情链接](#) [社区小黑板](#)