Menu 目录

一、参数详细说明

1.截取:

- 所有截取前后用&&连接,也可用\$\$(调试助手方式)
- 不使用&&表示用指定字符串,如使用固定标题,固定图片,"标题":"正片""图片":"http://xxx.xxx.com/uplod0034.jpg"
- 截取前可以使用通配符*, 如<a*>&&, 一个字段只能使用一个通配符

2.包含与不包含:

- 截取信息末尾可用中括号附加筛选包含或不包含,用#分隔多个关键词,全部英文符号,例
- "分类数组": "class=\"nav\"&&</div>[不包含:首页#资讯]",
- 数组筛选除关键词筛选外还可以筛选序号,连续序号用-连接序号之间用#分隔,例 "数组": "class=\"hl-list-item&&[不包含:1#9-11]",

3.替换:

- 可在截取信息末尾用中括号修改替换显示内容,替换与被替换之间用>>分隔,用#分隔多个替换项、替换为汉字空时表示删除被替换词
- 被替换词可以用通配符*,比如(*),会替换掉包含括号的左右括号之间的全部字符。替换内容也可以使用*,作用等同于&&截取,表示替换内容从网页获取,例 "线路数组": "class="hltabs-btn hl-slide-swiper&&[替换:线路1>>腾腾#播放>>空#(*)\$空]",// 这里的案例貌似有些问题
- 数组或列表的替换可以使用<序号>来从1开始按顺序编号,如[替换:v>>?ep=<序号>.mp4]

4.+拼接:

- +号拼接支持指定字符串与截取字符串混合拼接,如: /play/+/vod/&&.html+-1-1.html,表示把/vod/id编号.html拼接成/play/id编号-1-1.html
- 可无限拼接
- json模式的指定字符串拼接,需使用英文单引号,如: '/play/'+id+'-1-1.html',表示把从 json中获取的id值与前后字符串拼接

5.指定截取:

- 所有字段支持按分类指定截取
- 指定方式:默认--a&&b||连续剧--c&&d||首页--e&&f||搜索--g&&h,未指定的使用第一组,json方式也可以指定;
- 首页和搜索也可以指定截取,都在获取分类字段中填写,搜索字段为空或只有"搜索url"时, 搜索指定截取才生效
- 没有分类的单个文件或网页,可通过在地址后加?{cateld}(如果地址中已经有?,使用&),然后指定截取的方式进行分类展示
- 示例:

```
json
{
 "key": "live直播",
 "name": "live直播",
 "type": 3,
 "playerType": 1,
 "api": "csp_XBPQ",
 "searchable": 0,
 "quickSearch": 0,
 "filterable": 0,
 "ext": {
   "主页url": "./lives/live.txt",
   "直接播放": "1",
   "链接前缀": "http",
   "副标题": "group-title=\"&&\"",
   "分类": "央视$1#卫视$2#地方$3",
   //1,2,3随意编,只要不重复就行
   "分类url": "./lives/live.txt?{cateId}",
   "二次截取": "央视--\\#\\#央视&&\\#\\#||卫视--\\#\\#卫视&&\\#\\#||地方--\\#\\#<sup>‡</sup>
   //我在文件中手动加入##央视和##,作为央视列表的前后分割符,不影响直播,也能点播,因为#是
```

```
"数组": "INF:&&EXT",
"标题": ",&&http",
"图片": "http://",
"链接": "http&&\\#",
"嗅探词": "m3u8"
}
```

6.转义符:

XBPQ使用到的连接符(\$ # & * [])用于表示本义的时候,需要用\转义,比如:要截取 href="?cat&token=5543tdd5779fd87554gfy"中的token=5543tdd5779fd87554gfy, href="?cat\&&&"

7.json模式(不懂什么是json的忽略此模式):

- 对于json文档,可直接使用json方式,不需要截取
- 普通网页中的json数据,需要截取包括大括号{}在内的json数据
- 示例: data.list[1].name,表示获取json对象"data"的json数组"list"的第1个json对象的"name"属性值。
- json数组可指定下标,最小下标为1,如:list[](完整数组),list[n](list的第n个json对象),list[3,](不包含第1个第2个)
- 处于中间位置的json数组, []不能省略。

8.Base64:

整个html如果是Base64编码,可在二次截取处填写"Base64"表示不截取,只解码,如果截取数据是Base64编码,可使用Base64()解码,比如Base64(a&&b),表示对截取的结果进行Base64解码后再使用

9.填写及查错步骤

写规则最简步骤:

- ①填写"分类url", 大部分可以成功
- ②a.无分类展示数据:一般是分类获取错误,或者未指定数组情况下不能获取到有效数据。 分别尝试填写"分类"和"数组",如果还是没有数据,就把标题图片链接都填写完整。如果网站获取分类需要验证,直接放弃
- ②b.有分类, 点不开详情页: 这是"链接"获取错误, 补充填写正确的"链接"
- ②c.有分类,有详情,无播放列表或全部都是1: 这是播放列表获取错误,补充填写正确"播放数组"以及相关字段
- ②d.以上都有, 但无法播放: 这是获取"播放链接"错误, 补充填写正确"播放链接"及相关字段
- 可使用"调试":"1",来查看获取的链接是否是自己想要的结果

10.以下为完整字段

- "主页url": "",//默认从分类页中提取,一般不填
- "编码": "",//默认"UTF-8", 一般不填
- "首页": "",//可指定展示分类和数量,如"电影\$20",不指定数量默认展示最多40个,不指定分类默认展示首页推荐
- "请求头": "",//默认电脑ua, 特殊网站需填写为"手机"
- "头部集合": "",//支持头部数据集, 用\$和#分隔
- "播放请求头": "".//支持播放单独头部数据集. 用\$和#分隔
- "免嗅": "",//默认为0
- "嗅探词":"",//默认使用壳的规则
- "过滤词":"",//默认使用壳的规则
- "起始页": "",//默认1, 一般不填
- "直接播放": "",//默认0,为1时所有分类直接播放,也可以指定直接播放的分类,可以指定多个、以#分隔、比如"直接播放":"电影#首页"
- "短视频":"",//默认0,用于直接播放类,为1时,直接播放可以省略不写,会拉取整页短视频到播放列表,而不是单个播放,也可以按分类指定,比如"短视频":"直播"
- "强制解析":"",//默认0,全都走解析,解析失败再嗅探
- "倒序": "",//一般会自动倒序,不用填,也可强制指定
- "图片代理": "",//默认0,不显示图片时可试试1

10.1 获取大分类

- "分类url": "",//第一页与其他页不一样的(一般是第一页没有页码),直接用英文中括号加在分类url末尾,此处是否填写地区、剧情、年份、排序{area}、{class}、{year}、{by},决定是否开启相应筛选。不支持语言和字母{lang}、{letter}的筛选,最好不填写这个
- "分类": "",//用\$#分隔,海阔模式时以&分隔
- "分类值":"",//海阔模式时使用,以&分隔

10.2 也可以使用截取方式获取大分类

- "分类二次截取": "",
- "分类数组": "",
- "分类标题": "",
- "分类ID": "",

10.3 获取分类展示数据

- "二次截取": "", //二次截取常用于在数组之前,对页面进行过滤截取,常用于分类页面显示视频列表数据+推荐视频列表数据,可以通过二次截取,先获取视频列表数据范围,再通过分组只拿列表,不要推荐
- "数组": "",
- "图片": "",
- "标题": "",
- "副标题": "",
- "链接": "",
- "链接前缀": "",
- "链接后缀": "",

10.4 获取线路标题

- "线路二次截取": "",
- "线路数组": "",//可按线路名排序,在引号内末尾添加,如[排序:自建蓝光>腾腾>优优]
- "线路标题": "",

10.5 获取其他线路链接(用于播放列表不在详情页的情况,一般用不着)

- "多线二次截取": "",
- "多线数组": "",

- "多线链接": "",
- "多线链接前缀": "",
- "多线链接后缀": "",

10.6 获取详情

- "影片类型":"",
- "影片年代":"",
- "影片地区":"",
- "影片状态":"",
- "导演": "",
- "主演": "",
- "简介": "",

10.7 获取播放列表

- "播放二次截取": "",
- "播放数组": "", //播放数组匹配的结果应该是能匹配多个, 多条播放线路对应的列表数组, 而 截取是匹配一个, 里面包含了所有的数组内容
- "列表二次截取": "",
- "播放列表": "", // 一条播放线路对应的列表, 该线路下的所有分集列表
- "播放标题": "",
- "播放链接": "",
- "播放链接前缀": "",
- "播放链接后缀": "",

10.8 获取搜索数据

- 如果搜索全部不填,自动获取包括json和截取的搜索url并搜索;
- 如果只填了搜索url,会通过数组字段获取搜索数据,可单独指定搜索截取,指定方式:在其他 截取后追加||搜索--a&&b,json方式也可以;
- 如果填了搜索url和搜索数组,则通过搜索字段获取搜索数据。
- "搜索url": "",//可包含页码,会把搜索到的全部内容展示出来,jar能自动正确获取大部分网站搜索url,一般不用填写,手动post模式,网址;post;键1=值1&键2=值2

- "搜索模式": "",//默认json与截取混合使用
- "搜索二次截取": "",
- "搜索数组": "",
- "搜索图片": "",
- "搜索标题": "",
- "搜索链接": "",
- "搜索链接前缀": "",
- "搜索链接后缀": "",

10.9 获取筛选

- "类型": "",//对应{cteld}, 用\$#分隔, 海阔模式时以&分隔
- "类型值":"",//海阔模式时使用,以&分隔
- "剧情": "",//对应{class},用\$#分隔,海阔模式时以&分隔,内置通用模板,可使用[替换:惊悚>>空]的方式修改内置
- "剧情值":"",//海阔模式时使用,以&分隔
- "地区": "",//对应{area},用\$#分隔,海阔模式时以&分隔,内置通用模板,可使用[替换:大陆>>中国大陆]的方式修改内置
- "地区值":"",//海阔模式时使用,以&分隔
- "年份": "",//对应{year}, 默认筛选截止目前最近15年, 格式: 起始年-终止年, 也可使用海阔模式, 以&分隔
- "年份值":"",//海阔模式时使用,用&分隔
- "排序": "",//对应{by}, 默认英文模式, 时间\$time#热门\$hits, 以\$ #分隔
- "筛选": ""
- "筛选"不填写时,会根据"分类url"格式自动使用相应筛选
- 可在"筛选"字段填外部ison地址,支持本地clan和外网http

• 可直接复制包含大括号{}的ison数据到"筛选"字段,大括号前后不能有引号

二、例子

1.freeok

```
ison
{
   "key": "freeok",
   "name": "**1080P | FROK",
   "type": 3,
   "api": "csp_XBPQ",
   "ext": {
       "分类url": "https://www.freeok.vip/vod-show/{cateId}-{area}------
       "分类": "电影&剧集&动漫&综艺&短剧&少儿",
       "分类值": "1&2&3&4&12&5",
       "副标题": "<div class=\"module-item-note\">&</div>",
       "嗅探词": "m3u8#.m3u8#.mp4#freeok.mp4#/obj/",
       "线路数组": "data-dropdown-value=&&</div>[不包含:夸克#UC网盘]",
       "线路标题": "<span>&&</small>",
       "导演": "导演: &&</div>",
       "主演": "主演: &&</div>",
       "简介": "&&"
   }
}
```

2.完整案例

```
{"key": "黑狐","name": "黑狐 | BPQ","type": 3,"searchable": 1,"quickSearch": 1, "playerType": "2","api": "csp XBPQ",
"ext": {"站名": "黑狐影院",
'主页url": "aHR0cDovL2ZhZ21uLmNvbQ==",//Base64转码
"数组": "class=\"stui-vodlist_box\"&&</a",
"图片": "data-original=\"&&\"",
"标题": "title=\"&&\"",
"链接": "href=\"&&\"",
"嗅探词": ".m3u8#v3.huoshanvod.com#v3-dy-o.zjcdn#sf16-sg.larksuitecdn.com",
"过滤词": "/hls/#php",
"副标题": "<span class=\"pic-text text-right\">&&</span>",
"搜索url": "aHR0cDovL2ZhZ21uLmNvbQ==/search.php;post;searchword={wd}",
"搜索数组": "<a class=\"v-thumb stui-vodlist_thumb&&</a>",
"搜索图片": "data-original=\"&&\"",
"搜索标题": "title=\"&&\""
"搜索链接": "href=\"&&\""
"搜索副标题": "<span class=\"pic-text text-right\">&&</span>",
"搜索后缀": "/view/",
"线路数组": "href=\"#down&&/a>[替换:奇异视频>>黑狐一线#优酷视频>>黑狐二线#搜狐视频>>黑狐三线#咪咕视频>>黑狐四线)","
线路标题": ">&&<",
"播放数组": "class=\"stui-content_playlist clearfix&&",
"状态": "地区: &&",
"导演": "导演: &&
"主演": "主演: &&
"简介": "简介: &&</div>",
"播放列表": "<a&&/a>",
"播放标题": ">&&<",
"播放链接": "href=\"&&\"",
"免喧": "0".
"分类": "电影$1#电视剧$2#动漫$3#综艺$4",
"分类url": "aHR0cDovL2ZhZ21uLmNvbQ==/list/{cateId}_{catePg}.html?order=time" }},
```

以上代码基本是xbpq全规则,规则主要构成为数组数据截取、线路数据截取、搜索数据截取、播放数据截取。

本次主要介绍数组数据和播放数据的截取!

```
首先搞数组数据: "数组": "class="stui-vodlist__box"&</a",

"图片": "data-original="&&"",

"标题": "title="&&"",

"链接": "href="&&"",

"副标题": "<span class="pic-text text-right">&&</span>",
```

截取目的主要是找上面的代码在哪里?

打开电影网站<u>http://fagmn.com/</u>

点击电视剧或者电影分类-点右键,查看页面源代码。向下拉,随便找到一个影片的信息,如下:

```
164 "stui-pannel_bd">
165 ss="stui-vodlist clearfix">
166 class="col-md-7 col-sm-4 col-xs-3">
167 <div class="stui-vodlist__box">
       <a class="stui-vodlist__thumb lazyload" data-original="https://puui.q</pre>
168
       href="/edu-34520.html" title="达荷美女战士高清">
169
170
           <span class="play hidden-xs">
171
           </span>
           <span class="pic-text text-right">
173
               高清
174
           </span>
175
       </a>
176
       <div class="stui-vodlist__detail">
177
           <h4 class="title text-overflow">
178
               <a href="/edu-34520.html" title="达荷美女战士">
179
                  达荷美女战士
               </a>
180
181
           </h4>
           维奥拉·戴维斯 图索·姆贝杜 拉什纳·林奇 希拉·
183
184
           185
       </div>
186 </div>
187 i>
```

按照上面代码的显示把它翻译到规则代码里,注意""要用\来转译,例如"标题":"title="&&"",里面的\就是源码里""的转译!

&& 就是你要取的内容 例如标题的内容 手工少女*****

接下来就一一对应截取下来!

照葫芦画瓢, 多搞几次就会了。

播放数组:随便点开一个影片,点右键,查看页面源代码。向下拉,随便找到一个影片的信息,和数组一样,照葫芦画瓢。

搜索数组:找到搜索,随便搜个影片,点右键,查看页面源代码。向下拉,随便找到一个影片的信息,和数组一样,照葫芦画瓢。

线路数据:同播放数组

使用海阔世界的源码助手可以帮助快速的精简代码。

https://haikuo.lanzouf.com/u/GoldRiver

三、精缝

- 1,准备XBPQ.jar的包,约300k大小,<u>XBPQ.jar</u>
- 2,解压缩获取classes.dex。用MT管理器将dex文件打开

- 3,将spider文件夹内的所有文件,长按以后,选择重命名(XBPQ除外),改成xxx-xbpq。主要是防止除XBPQ外的其它文件与已有文件名冲突。
- 4,将spider内的merge文件夹,长按后重命名为merge-xbpg,也是为了防止冲突。
- 5,修改完成后选择退出merge并保持修改。
- 6, 打开classes.dex时,选择转成smail。得到classes.zip压缩文件。
- 7,解压缩classes.zip文件,如果遇到解压缩文件冲突,选择修改文件名以共存。
- 8,将解压缩后的文件夹内的spider和parser文件夹,复制到CatVodSpider项目的./jar/3rd/xbpg/目录内。
- 9, 修改genJar.bat文件, 在复制smail文件时, 将上一步的文件夹复制进去, 达到缝合目的

@echo off

del "%~dp0\custom_spider.jar"
 java -jar "%~dp0\3rd\baksmali-2.5.2.jar" d "%~dp0\..\app\build\intermediates\
 java -jar "%~dp0\3rd\XBPQ.jar" d "%~dp0\..\app\build\intermediates\dex\releas
 rd /s/q "%~dp0\spider.jar\smali\com\github\catvod\spider"
 rd /s/q "%~dp0\spider.jar\smali\com\github\catvod\parser"
 rd /s/q "%~dp0\spider.jar\smali\com\github\catvod\js"
 if not exist "%~dp0\spider.jar\smali\com\github\catvod\" md "%~dp0\spider.jar
move "%~dp0\Smali_classes\com\github\catvod\spider" "%~dp0\spider.jar\smali\c
move "%~dp0\Smali_classes\com\github\catvod\parser" "%~dp0\spider.jar\smali\c
move "%~dp0\Smali_classes\com\github\catvod\js" "%~dp0\spider.jar\smali\com\g

xcopy /s "%~dp0\3rd\xbpq\spider\" "%~dp0\spider.jar\smali\com\github\catvod\s
if not exist "%~dp0\spider.jar\smali\com\github\catvod\parser\" md "%~dp0\spi
xcopy /s "%~dp0\3rd\xbpq\parser\" "%~dp0\spider.jar\smali\com\github\catvod\p

```
java -jar "%~dp0\3rd\apktool_2.4.1.jar" b "%~dp0\spider.jar" -c
move "%~dp0\spider.jar\dist\dex.jar" "%~dp0\custom_spider.jar"
certUtil -hashfile "%~dp0\custom_spider.jar" MD5 | find /i /v "md5" | find /i
rd /s/q "%~dp0\spider.jar\build"
rd /s/q "%~dp0\spider.jar\smali"
rd /s/q "%~dp0\spider.jar\dist"
rd /s/q "%~dp0\Smali_classes"
```

上述文件新增了两个xcopy的命令,就是两个空行之间的代码。

Previous page maccms搭建

10,运行build.bat打包,打包过程与原打包相同。

Next page drpyJS写源教程