ctriptech 发布于 携程技术中心
2016年06月30日 · 50.8k 次阅读

关于反爬虫，看这一篇就够了

编者：本文来自携程酒店研发部研发经理崔广宇在第三期【携程技术微分享】上的分享，以下为整理的内容概要。墙裂建议点击视频回放，“现场”围观段子手攻城狮大崔，如何高智商&高情商地完美碾压爬虫。。。关注携程技术中心微信公号ctriptech，可第一时间获知微分享信息~

你被爬虫侵扰过么？当你看到“爬虫”两个字的时候，是不是已经有点血脉贲张的感觉了？千万要忍耐，稍稍做点什么，就可以在名义上让他们胜利，实际上让他们受损失。

一、为什么要反爬虫

1、爬虫占总PV比例较高，这样浪费钱（尤其是三月份爬虫）。

三月份爬虫是个什么概念呢？每年的三月份我们会迎接一次爬虫高峰期。

最初我们百思不得其解。直到有一次，四月份的时候，我们删除了一个url，然后有个爬虫不断的爬取url，导致大量报错，测试开始找我们麻烦。我们只好特意为这个爬虫发布了一次站点，把删除的url又恢复回去了。

但是当时我们的一个组员表示很不服，说，我们不能干掉爬虫，也就罢了，还要专门为它发布，这实在是太没面子了。于是出了个主意，说：url可以上，但是，绝对不给真实数据。

于是我们就把一个静态文件发布上去了。报错停止了，爬虫没有停止，也就是说对方并不知道东西都是假的。这个事情给了我们一个很大的启示，也直接成了我们反爬虫技术的核心：变更。

后来有个学生来申请实习。我们看了简历发现她爬过携程。后来面试的时候确认了下，果然她就是四月份害我们发布的那个家伙。不过因为是个妹子，技术也不错，后来就被我们招安了。现在已经快正式入职了。

后来我们一起讨论的时候，她提到了，有大量的硕士在写论文的时候会选择爬取OTA数据，并进行舆情分析。因为五月份交论文，所以嘛，大家都是读过书的，你们懂的，前期各种DotA，LOL，到了三月份

就是这么个节奏。

2、公司可免费查询的资源被批量抓走，丧失竞争力，这样少赚钱。

OTA的价格可以在非登录状态下直接被查询，这个是底线。如果强制登陆，那么可以通过封杀账号的方式让对方付出代价，这也是很多网站的做法。但是我们不能强制对方登录。那么如果没有反爬虫，对方就可以批量复制我们的信息，我们的竞争力就会大大减少。

竞争对手可以抓到我们的价格，时间长了用户就会知道，只需要去竞争对手那里就可以了，没必要来携程。这对我们是不利的。

3、爬虫是否涉嫌违法？ 如果是的话，是否可以起诉要求赔偿？这样可以赚钱。

这个问题我特意咨询了法务，最后发现这在国内还是个擦边球，就是有可能可以起诉成功，也可能完全无效。所以还是需要用技术手段来做最后的保障。

二、反什么样的爬虫

1、十分低级的应届毕业生

开头我们提到的三月份爬虫，就是一个十分明显的例子。应届毕业生的爬虫通常简单粗暴，根本不管服务器压力，加上人数不可预测，很容易把站点弄挂。

顺便说下，通过爬携程来获取offer这条路已经行不通了。因为我们都知道，第一个说漂亮女人像花的人，是天才。而第二个。。。你们懂的吧?

2、十分低级的创业小公司

现在的创业公司越来越多，也不知道是被谁忽悠的然后大家创业了发现不知道干什么好，觉得大数据比较热，就开始做大数据。

分析程序全写差不多了，发现自己手头没有数据。

怎么办？写爬虫爬啊。于是就有了不计其数的小爬虫，出于公司生死存亡的考虑，不断爬取数据。

3、不小心写错了没人去停止的失控小爬虫

携程上的点评有的时候可能高达60%的访问量是爬虫。我们已经选择直接封锁了，它们依然孜孜不倦地爬取。

什么意思呢？就是说，他们根本爬不到任何数据，除了http code是200以外，一切都是不对的，可是爬虫依然不停止这个很可能就是一些托管在某些服务器上的小爬虫，已经无人认领了，依然在辛勤地工作着。

4、成型的商业对手

这个是最大的对手，他们有技术，有钱，要什么有什么，如果和你死磕，你就只能硬着头皮和他死磕。

5、抽风的搜索引擎

大家不要以为搜索引擎都是好人，他们也有抽风的时候，而且一抽风就会导致服务器性能下降，请求量跟网络攻击没什么区别。

三、什么是爬虫和反爬虫

因为反爬虫暂时是个较新的领域，因此有些定义要自己下。我们内部定义是这样的：

- 爬虫：使用任何技术手段，批量获取网站信息的一种方式。关键在于批量。
- 反爬虫：使用任何技术手段，阻止别人批量获取自己网站信息的一种方式。关键也在于批量。
- 误伤：在反爬虫的过程中，错误的将普通用户识别为爬虫。误伤率高的反爬虫策略，效果再好也不能用。
- 拦截：成功地阻止爬虫访问。这里会有拦截率的概念。通常来说，拦截率越高的反爬虫策略，误伤的可能性就越高。因此需要做个权衡。
- 资源：机器成本与人力成本的总和。

这里要切记，人力成本也是资源，而且比机器更重要。因为，根据摩尔定律，机器越来越便宜。而根据IT行业的发展趋势，程序员工资越来越贵。因此，让对方加班才是王道，机器成本并不是特别值钱。

四、知己知彼：如何编写简单爬虫

要想做反爬虫，我们首先需要知道如何写个简单的爬虫。

目前网络上搜索到的爬虫资料十分有限，通常都只是给一段python代码。python是一门很好的语言，但是用来针对有反爬虫措施的站点做爬虫，真的不是最优选择。

更讽刺的是，通常搜到的python爬虫代码都会使用一个lynx的user-agent。你们应该怎么处理这个user-agent，就不用我来说了吧？

- 分析页面请求格式
- 创建合适的http请求
- 批量发送http请求，获取数据

举个例子，直接查看携程生产url。在详情页点击“确定”按钮，会加载价格。假设价格是你想要的，那么抓出网络请求之后，哪个请求才是你想要的结果呢?

答案出乎意料的简单，你只需要用根据网络传输数据量进行倒序排列即可。因为其他的迷惑性的url再多再复杂，开发人员也不会舍得加数据量给他。

五、知己知彼：如何编写高级爬虫

那么爬虫进阶应该如何做呢？通常所谓的进阶有以下几种：

分布式

通常会有一些教材告诉你，为了爬取效率，需要把爬虫分布式部署到多台机器上。这完全是骗人的。分布式唯一的作用是：防止对方封IP。封IP是终极手段，效果非常好，当然，误伤起用户也是非常爽的。

模拟JavaScript

有些教程会说，模拟javascript，抓取动态网页，是进阶技巧。但是其实这只是个很简单的功能。因为，如果对方没有反爬虫，你完全可以直接抓ajax本身，而无需关心js怎么处理的。如果对方有反爬虫，那么javascript必然十分复杂，重点在于分析，而不仅仅是简单的模拟。

换句话说：这应该是基本功。

PhantomJs

这个是一个极端的例子。这个东西本意是用来做自动测试的，结果因为效果很好，很多人拿来做爬虫。但是这个东西有个硬伤，就是：效率。此外PhantomJs也是可以被抓到的，出于多方面原因，这里暂时不讲。

六、不同级别爬虫的优缺点

越是低级的爬虫，越容易被封锁，但是性能好，成本低。越是高级的爬虫，越难被封锁，但是性能低，成本也越高。

当成本高到一定程度，我们就可以无需再对爬虫进行封锁。经济学上有个词叫边际效应。付出成本高到一

那么如果对双方资源进行对比，我们就会发现，无条件跟对方死磕，是不划算的。应该有个黄金点，超过这个点，那就让它爬好了。毕竟我们反爬虫不是为了面子，而是为了商业因素。

七、如何设计一个反爬虫系统(常规架构)

有个朋友曾经给过我这样一个架构：

1、对请求进行预处理，便于识别；
2、识别是否是爬虫；
3、针对识别结果，进行适当的处理；

当时我觉得，听起来似乎很有道理，不愧是架构，想法就是和我们不一样。后来我们真正做起来反应过来不对了。因为：

如果能识别出爬虫，哪还有那么多废话？想怎么搞它就怎么搞它。如果识别不出来爬虫，你对谁做适当处理？

三句话里面有两句是废话，只有一句有用的，而且还没给出具体实施方式。那么：这种架构(师)有什么用？

因为当前存在一个架构师崇拜问题，所以很多创业小公司以架构师名义招开发。给出的title都是：初级架构师，架构师本身就是个高级岗位，为什么会有初级架构。这就相当于：初级将军/初级司令。

最后去了公司，发现十个人，一个CTO，九个架构师，而且可能你自己是初级架构师，其他人还是高级架构师。不过初级架构师还不算坑爹了，有些小创业公司还招CTO做开发呢。

传统反爬虫手段

1、后台对访问进行统计，如果单个IP访问超过阈值，予以封锁。

这个虽然效果还不错，但是其实有两个缺陷，一个是非常容易误伤普通用户，另一个就是，IP其实不值钱，几十块钱甚至有可能买到几十万个IP。所以总体来说是比较亏的。不过针对三月份呢爬虫，这点还是非常有用的。

2、后台对访问进行统计，如果单个session访问超过阈值，予以封锁。

这个看起来更高级了一些，但是其实效果更差，因为session完全不值钱，重新申请一个就可以了。

3、后台对访问进行统计，如果单个userAgent访问超过阈值，予以封锁。

这个是大招，类似于抗生素之类的，效果出奇的好，但是杀伤力过大，误伤非常严重，使用的时候要非常小心。至今为止我们也就只短暂封杀过mac下的火狐。

4、以上的组合

组合起来能力变大，误伤率下降，在遇到低级爬虫的时候，还是比较好用的。

由以上我们可以看出，其实爬虫反爬虫是个游戏，RMB玩家才最牛逼。因为上面提到的方法，效果均一般，所以还是用JavaScript比较靠谱。

也许有人会说：javascript做的话，不是可以跳掉前端逻辑，直接拉服务吗？怎么会靠谱呢？因为啊，我是一个标题党啊。JavaScript不仅仅是做前端。跳过前端不等于跳过JavaScript。也就是说：我们的服务器是nodejs做的。

> 思考题：我们写代码的时候，最怕碰到什么代码？什么代码不好调试？

eval

eval已经臭名昭著了，它效率低下，可读性糟糕。正是我们所需要的。

goto

js对goto支持并不好，因此需要自己实现goto。

混淆

目前的minify工具通常是minify成abcd之类简单的名字，这不符合我们的要求。我们可以minify成更好用的，比如阿拉伯语。为什么呢？ 因为阿拉伯语有的时候是从左向右写，有的时候是从右向左写，还有的时候是从下向上写。除非对方雇个阿拉伯程序员，否则非头疼死不可。

不稳定代码

什么bug不容易修？不容易重现的bug不好修。因此，我们的代码要充满不确定性，每次都不一样。

代码演示

下载代码本身，可以更容易理解。这里简短介绍下思路：

1. 纯JAVASCRIPT反爬虫DEMO，通过更改连接地址，来让对方抓取到错误价格。这种方法简单，但

2. 纯JAVASCRIPT反爬虫DEMO，更改key。这种做法简单，不容易被发现。但是可以通过有意爬取错误价格的方式来实现。
3. 纯JAVASCRIPT反爬虫DEMO，更改动态key。这种方法可以让更改key的代价变为0，因此代价更低。
4. 纯JAVASCRIPT反爬虫DEMO，十分复杂的更改key。这种方法，可以让对方很难分析，如果加了后续提到的浏览器检测，更难被爬取。

到此为止。

前面我们提到了边际效应，就是说，可以到此为止了。后续再投入人力就得不偿失了。除非有专门的对手与你死磕。不过这个时候就是为了尊严而战，不是为了商业因素了。

浏览器检测

针对不同的浏览器，我们的检测方式是不一样的。

- IE 检测bug；
- FF 检测对标准的严格程度；
- Chrome 检测强大特性。

八、我抓到你了——然后该怎么办

不会引发生产事件——直接拦截

可能引发生产事件——给假数据(也叫投毒)

此外还有一些发散性的思路。例如是不是可以在响应里做SQL注入？毕竟是对方先动的手。不过这个问题法务没有给具体回复，也不容易和她解释。因此暂时只是设想而已。

1、技术压制

我们都知道，DotA AI里有个de命令，当AI被击杀后，它获取经验的倍数会提升。因此，前期杀AI太多，AI会一身神装，无法击杀。

正确的做法是，压制对方等级，但是不击杀。反爬虫也是一样的，不要一开始就搞太过分，逼人家和你死磕。

2、心理战

以上略过不提，大家领会精神即可。

3、放水

这个可能是是最高境界了。

程序员都不容易，做爬虫的尤其不容易。可怜可怜他们给他们一小口饭吃吧。没准过几天你就因为反爬虫做得好，改行做爬虫了。

比如，前一阵子就有人找我问我会不会做爬虫。。。。。我这么善良的人，能说不会吗？？？？

你可能感兴趣的文章

- **手把手教你写电商爬虫-第五课 京东商品评论爬虫 一起来对付反爬虫** 爬虫教程 网页爬虫 大数据 电商网站 python javascript
- **Python 从零开始爬虫(一)——爬虫伪装&反“反爬”** β_3000 网页爬虫 python
- **互联网金融爬虫怎么写－第一课 p2p网贷爬虫（XPath入门）** 爬虫教程 前端 爬虫图片 网页爬虫 python javascript
- **手把手教你写电商爬虫-第三课 实战尚妆网AJAX请求处理和内容提取** 爬虫教程 github 网页爬虫 电商网站 python javascript
- **互联网金融爬虫怎么写－第二课 雪球网股票爬虫（正则表达式入门）** 爬虫教程 前端 爬虫图片 网页爬虫 python javascript
- **手把手教你写电商爬虫-第四课 淘宝网商品爬虫自动JS渲染** 爬虫教程 大数据 网页爬虫 电商 python javascript
- **node.js 爬取招聘信息分析各职业钱途（爬虫+动态IP代理+数据可视化分析）** 微雨微语 javascript 前端 node.js 网页爬虫 es6
- **nodeJS实现基于Promise爬虫 定时发送信息到指定邮件** lucas_580e331d326b4 javascript nodejs爬虫 node.js 前端 程序员

31 条评论

默认排序 | 时间排序