

03 | HTTP世界全览（上）：与HTTP相关的各种概念

2019-06-03 Chrono

《透视HTTP协议》

课程介绍 >



讲述：Chrono

时长 10:46 大小 12.34M



在上一讲的末尾，我画了一张图，里面是与 HTTP 关联的各种技术和知识点，也可以说是这个专栏的总索引，不知道你有没有认真看过呢？

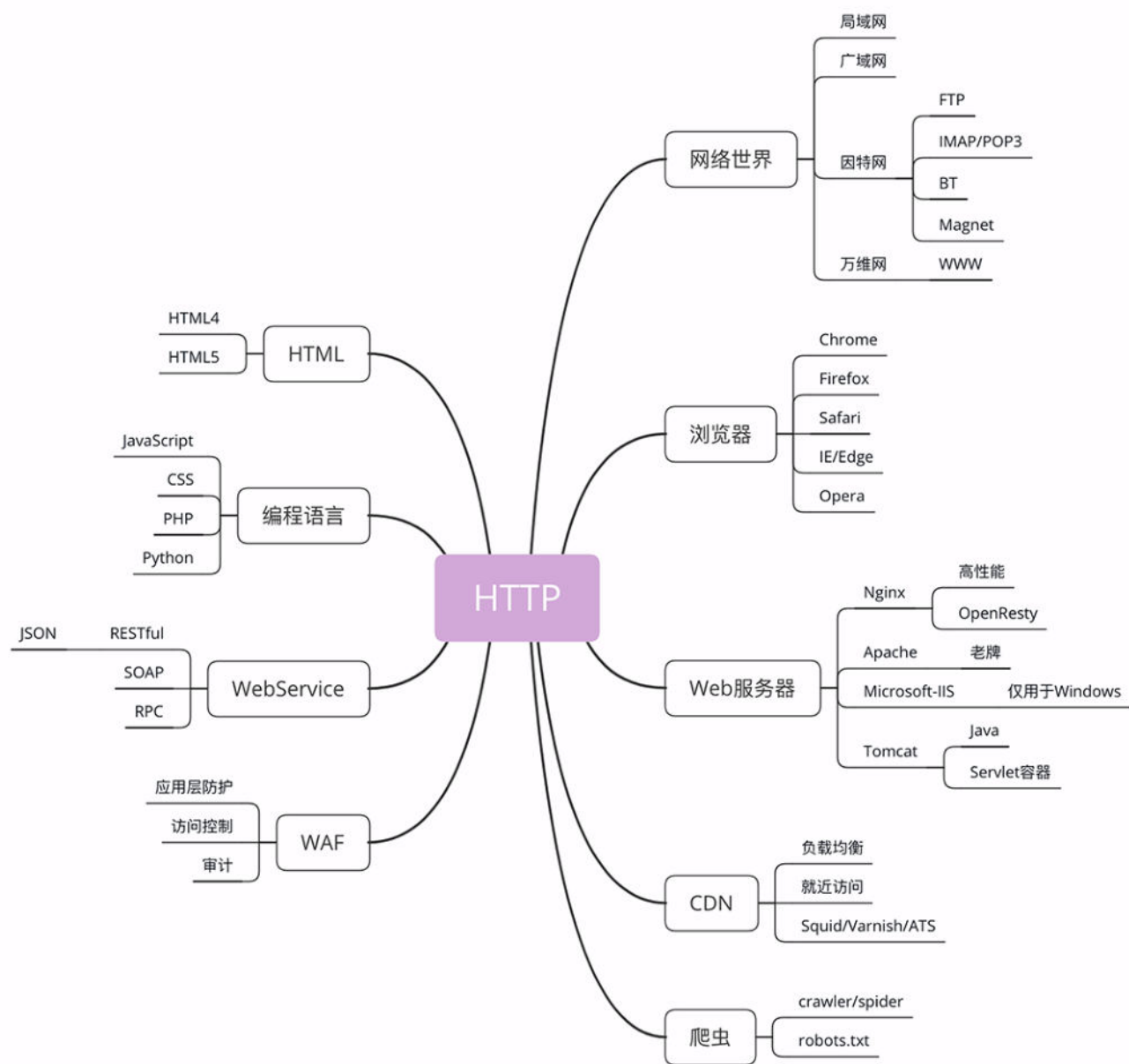
那张图左边的部分是与 HTTP 有关系的各种协议，比较偏向于理论；而右边的部分是与 HTTP 有关系的各种应用技术，偏向于实际应用。

我希望借助这张图帮你澄清与 HTTP 相关的各种概念和角色，让你在实际工作中清楚它们在链路中的位置和作用，知道发起一个 HTTP 请求会有哪些角色参与，会如何影响请求的处理，做到“手中有粮，心中不慌”。

领资料

因为那张图比较大，所以我会把左右两部分拆开来分别讲，今天先讲右边的部分，也就是与 HTTP 相关的各种应用，着重介绍互联网、浏览器、Web 服务器等常见且重要的概念。





为了方便你查看，我又把这部分重新画了一下，比那张大图小了一些，更容易地阅读，你可以[点击查看](#)。

暖场词就到这里，让我们正式开始吧。

网络世界

你一定已经习惯了现在的网络生活，甚至可能会下意识地认为网络世界就应该是这个样子的：“一张平坦而且一望无际的巨大网络，每一台电脑就是网络上的一个节点，均匀地点缀在这张网上”。

领资料



这样的理解既对，又不对。从抽象的、虚拟的层面来看，网络世界确实是这样的，我们可以从一个节点毫无障碍地访问到另一个节点。

但现实世界的网络却远比这个抽象的模型要复杂得多。实际的互联网是由许许多多多个规模略小的网络连接而成的，这些“小网络”可能是只有几百台电脑的局域网，可能是有几万、几十万台电脑的广域网，可能是用电缆、光纤构成的固定网络，也可能是用基站、热点构成的移动网络.....

互联网世界更像是由数不清的大小岛屿组成的“千岛之国”。

互联网的正式名称是 Internet，里面存储着无穷无尽的信息资源，我们通常所说的“上网”实际上访问的只是互联网的一个子集“万维网”（World Wide Web），它基于 HTTP 协议，传输 HTML 等超文本资源，能力也就被限制在 HTTP 协议之内。

互联网上还有许多万维网之外的资源，例如常用的电子邮件、BT 和 Magnet 点对点下载、FTP 文件下载、SSH 安全登录、各种即时通信服务等等，它们需要用各自的专有协议来访问。

不过由于 HTTP 协议非常灵活、易于扩展，而且“超文本”的表述能力很强，所以很多其他原本不属于 HTTP 的资源也可以“包装”成 HTTP 来访问，这就是我们为什么能够总看到各种“网页应用”——例如“微信网页版”“邮箱网页版”——的原因。

综合起来看，现在的互联网 90% 以上的部分都被万维网，也就是 HTTP 所覆盖，所以把互联网约等于万维网或 HTTP 应该也不算大错。

浏览器

上网就要用到浏览器，常见的浏览器有 Google 的 Chrome、Mozilla 的 Firefox、Apple 的 Safari、Microsoft 的 IE 和 Edge，还有小众的 Opera 以及国内的各种“换壳”的“极速”“安全”浏览器。

领资料





那么你想过没有，所谓的“浏览器”到底是个什么东西呢？

浏览器的正式名字叫“**Web Browser**”，顾名思义，就是检索、查看互联网上网页资源的应用程序，名字里的 Web，实际上指的就是“World Wide Web”，也就是万维网。

浏览器本质上是一个 HTTP 协议中的**请求方**，使用 HTTP 协议获取网络上的各种资源。当然，为了让我们更好地检索查看网页，它还集成了很多额外的功能。

例如，HTML 排版引擎用来展示页面，JavaScript 引擎用来实现动态化效果，甚至还有开发者工具用来调试网页，以及五花八门的各种插件和扩展。

在 HTTP 协议里，浏览器的角色被称为“User Agent”即“用户代理”，意思是作为访问者的“代理”来发起 HTTP 请求。不过在不引起混淆的情况下，我们通常都简单地称之为“客户端”。

Web 服务器

刚才说的浏览器是 HTTP 里的请求方，那么在协议另一端的**应答方**（响应方）又是什么呢？

这个你一定也很熟悉，答案就是**服务器**，**Web Server**。

Web 服务器是一个很大很重要的概念，它是 HTTP 协议里响应请求的主体，通常也把控着绝大多数的网络资源，在网络世界里处于强势地位。

当我们谈到“Web 服务器”时有两个层面的含义：硬件和软件。

硬件含义就是物理形式或“云”形式的机器，在大多数情况下它可能不是一台服务器，而是利用反向代理、负载均衡等技术组成的庞大集群。但从外界看来，它仍然表现为一台机器，但这个形象是“虚拟的”。

领资料



软件含义的 Web 服务器可能我们更为关心，它就是提供 Web 服务的应用程序，通常会运行在硬件含义的服务器上。它利用强大的硬件能力响应海量的客户端 HTTP 请求，处理磁盘上的网页、图片等静态文件，或者把请求转发给后面的 Tomcat、Node.js 等业务应用，返回动态的信息。

比起层出不穷的各种 Web 浏览器，Web 服务器就要少很多了，一只手的手指头就可以数得过来。

Apache 是老牌的服务器，到今天已经快 25 年了，功能相当完善，相关的资料很多，学习门槛低，是许多创业者建站的入门产品。

Nginx 是 Web 服务器里的后起之秀，特点是高性能、高稳定，且易于扩展。自 2004 年推出后就不断蚕食 Apache 的市场份额，在高流量的网站里更是不二之选。

此外，还有 Windows 上的 IIS、Java 的 Jetty/Tomcat 等，因为性能不是很高，所以在互联网上应用得较少。

CDN

浏览器和服务是 HTTP 协议的两个端点，那么，在这两者之间还有别的什么东西吗？

当然有了。浏览器通常不会直接连到服务器，中间会经过“重重关卡”，其中的一个重要角色就叫做 CDN。

CDN，全称是“Content Delivery Network”，翻译过来就是“内容分发网络”。它应用了 HTTP 协议里的缓存和代理技术，代替源站响应客户端的请求。

CDN 有什么好处呢？

简单来说，它可以缓存源站的数据，让浏览器的请求不用“千里迢迢”地到达源站服务器，直接在“半路”就可以获取响应。如果 CDN 的调度算法很优秀，更可以找到离用户最近的节点，大幅度缩短响应时间。

打个比方，就好像唐僧西天取经，刚出长安城，就看到阿难与迦叶把佛祖的真经递过来了，是不是很省事？

领资料



CDN 也是现在互联网中的一项重要基础设施，除了基本的网络加速外，还提供负载均衡、安全防护、边缘计算、跨运营商网络等功能，能够成倍地“放大”源站服务器的服务能力，很多云服务商都把 CDN 作为产品的一部分，我也会在后面用一讲的篇幅来专门讲解 CDN。

爬虫

前面说到过浏览器，它是一种用户代理，代替我们访问互联网。

但 HTTP 协议并没有规定用户代理后面必须是“真正的人类”，它也完全可以是“机器人”，这些“机器人”的正式名称就叫做“**爬虫**”（Crawler），实际上是一种可以自动访问 Web 资源的应用程序。

“爬虫”这个名字非常形象，它们就像是一只只不知疲倦的、辛勤的蚂蚁，在无边无际的网络上爬来爬去，不停地在网站间奔走，搜集抓取各种信息。

据估计，互联网上至少有 50% 的流量都是由爬虫产生的，某些特定领域的比例还会更高，也就是说，如果你的网站今天的访问量是十万，那么里面至少有五六万是爬虫机器人，而不是真实的用户。

爬虫是怎么来的呢？

绝大多数是由各大搜索引擎“放”出来的，抓取网页存入庞大的数据库，再建立关键字索引，这样我们才能够在搜索引擎中快速地搜索到互联网角落里的页面。

爬虫也有不好的一面，它会过度消耗网络资源，占用服务器和带宽，影响网站对真实数据的分析，甚至导致敏感信息泄漏。所以，又出现了“反爬虫”技术，通过各种手段来限制爬虫。其中一项就是“君子协定”robots.txt，约定哪些该爬，哪些不该爬。

无论是“爬虫”还是“反爬虫”，用到的基本技术都是两个，一个是 HTTP，另一个就是 HTML。

HTML/WebService/WAF

到现在我已经说完了图中右边的五大部分，而左边的 HTML、WebService、WAF 等由于与 HTTP 技术上实质关联不太大，所以就简略地介绍一下，不再过多展开。

领资料



HTML 是 HTTP 协议传输的主要内容之一，它描述了超文本页面，用各种“标签”定义文字、图片等资源和排版布局，最终由浏览器“渲染”出可视化页面。

HTML 目前有两个主要的标准，HTML4 和 HTML5。广义上的 HTML 通常是指 HTML、JavaScript、CSS 等前端技术的组合，能够实现比传统静态页面更丰富的动态页面。

接下来是 **Web Service**，它的名字与 Web Server 很像，但却是一个完全不同的东西。

Web Service 是一种由 W3C 定义的应用服务开发规范，使用 client-server 主从架构，通常使用 WSDL 定义服务接口，使用 HTTP 协议传输 XML 或 SOAP 消息，也就是说，它是一个**基于 Web（HTTP）的服务架构技术**，既可以运行在内网，也可以在适当保护后运行在外网。

因为采用了 HTTP 协议传输数据，所以在 Web Service 架构里服务器和客户端可以采用不同的操作系统或编程语言开发。例如服务器端用 Linux+Java，客户端用 Windows+C#，具有跨平台跨语言的优点。

WAF 是近几年比较“火”的一个词，意思是“网络应用防火墙”。与硬件“防火墙”类似，它是应用层面的“防火墙”，专门检测 HTTP 流量，是防护 Web 应用的安全技术。

WAF 通常位于 Web 服务器之前，可以阻止如 SQL 注入、跨站脚本等攻击，目前应用较多的一个开源项目是 ModSecurity，它能够完全集成进 Apache 或 Nginx。

小结

今天我详细介绍了与 HTTP 有关系的各种应用技术，在这里简单小结一下要点。

1. 互联网上绝大部分资源都使用 HTTP 协议传输；
2. 浏览器是 HTTP 协议里的请求方，即 User Agent；
3. 服务器是 HTTP 协议里的应答方，常用的有 Apache 和 Nginx；
4. CDN 位于浏览器和服务器之间，主要起到缓存加速的作用；
5. 爬虫是另一类 User Agent，是自动访问网络资源的程序。

希望通过今天的讲解，你能够更好地理解这些概念，也利于后续的课程学习。

领资料



课下作业

1. 你觉得 CDN 在对待浏览器和爬虫时会有差异吗？为什么？
2. 你怎么理解 Webservice 与 Web Server 这两个非常相似的词？

欢迎你通过留言分享答案，与我和其他同学一起讨论。如果你觉得有所收获，欢迎你把文章分享给你的朋友。



== 课外小贴士 ==

- 01 第一个网页浏览器也是蒂姆·伯纳斯-李发明的，名字就叫 WorldWideWeb。
- 02 第一个 Web 服务器由蒂姆·伯纳斯-李设计并参与开发，名字叫 CERN httpd。
- 03 Linux 上的 wget、curl 等命令行工具基于 http，所以也是一种 user agent。
- 04 Nginx 的正确发音应该是“engine eks”，不过我更愿意像 UNIX/Linux 那样称它为“engine ks”——虽然这是一个“错误”的发音，但却简洁明快。


领资料



05 你也许在浏览某些网站时遇到过要求“验证你不是机器人”的页面，这其实就是一种“反爬虫”手段。

分享给需要的人，Ta订阅超级会员，你将得 50 元

Ta单独购买本课程，你将得 20 元

 生成海报并分享

 赞 40

 提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 02 | HTTP是什么？HTTP又不是什么？

下一篇 04 | HTTP世界全览（下）：与HTTP相关的各种协议

领资料



JVM + NIO + Spring

各大厂面试题及知识点详解

限时免费



精选留言 (79)

写留言



Shopee内推码: NTAGx...

2019-06-05

1. CDN 应当是不区分的, 因为爬虫本身也是对 Web 资源的访问, 且对于爬虫识别并不是 100% 准确的, 因此 CDN 只会去计算实际使用了多少资源而不管其中多少来自爬虫;
2. 个人理解, Web Service 是网络服务实体, 而 Web Server 是网络服务器, 后者的存在是为了承载前者。

作者回复: ✓

共 2 条评论 >

64



Amark

2019-06-03

老师, 能不能通俗地讲讲RPC, SOAP, restful, 之间的区别

作者回复: 这个话题比较大。rpc就是把网络通信封装成了函数调用的形式, 所以叫rpc。soap是web service的消息格式。RESTful是一种web服务接口的设计理念。

这三章都是与应用服务有关, 但领域不同。

共 2 条评论 >

28

领资料





飒~

2019-06-12

老师，暗网是如何规避搜索引擎的爬虫的，它又是怎么被人访问的呢

作者回复: 这个问题比较高端，有其他知道的同学吗？

共 6 条评论 >

👍 26



Vickey Cheung

2019-09-09

老师，web服务器和web容器区别是什么呢？

作者回复: web服务器主要提供静态资源，而web容器可以运行Java、php等程序提供动态服务。



👍 12



耿斌

2019-06-06

1. CDN可以根据User-Agent来判断发起请求的一端是浏览器还是爬虫，对待爬虫可以特殊处理返回特定内容
2. WebService是基于Web（HTTP）的服务器架构技术，基于HTTP协议传输xml或soap数据。WebServer分硬件和软件，硬件指服务器、云之类，软件如Nginx、Apache等

作者回复: √

共 4 条评论 >

👍 11



redrain

2019-06-03

有些网站全新上线的，没有外链，也没特意提交过，为什么也会有爬虫经过呢，入口在哪里

作者回复: 这可能是从dns域名服务商那里获取了你的网站。



👍 11

领资料



钱

2020-03-23

首先，非常喜欢这种评论多，互动多的专栏，好像买一赠一，感觉赚啦！有时有些疑惑看完评论就想明白了，这也是我判断一个专栏优质与否的标准之一，老师的专栏很棒，点赞！

1: 你觉得 CDN 在对待浏览器和爬虫时会有差异吗？为什么？



刚开始觉得应该有区分的，不然反爬虫技术是干什么吃的，后来想想如果不用反爬虫技术的话应该是没差异的，因为CDN的核心工作是把网络的静态资源放的离用户更近一些，加速网络信息的获取速度，那区分是人要的信息还是机器人要的信息意义也不大，关键是我觉得爬虫如果做的好模仿真人来获取信息，那CND也是很难区分的，再者静态资源本来也是公开透明让用户代理来访问的嘛！

2：你怎么理解 WebService 与 Web Server 这两个非常相似的词？

首先，Web Server比较容易理解，就是web服务器，有软的也有硬的，软的特指有程序代码实现，硬的特指有实实在在的硬件机器组成，比如：Apache、nginx、tomcat、jetty等。
web service 直译是web服务，不过这么讲还是比较抽象不知道她是什么玩意？只好找一下，她的定义了。

Web Service 是一种由 W3C 定义的应用服务开发规范，使用 client-server 主从架构，通常使用 WSDL 定义服务接口，使用 HTTP 协议传输 XML 或 SOAP 消息，也就是说，它是一个基于 Web (HTTP) 的服务架构技术，既可以运行在内网，也可以在适当保护后运行在外网。
OK，直白来说，web service 是一种开发规范，规范即约定稍微有些强制执行的意味。她和 web Server 完全不是一种类型的东西，她比较虚是一个组织对某些行为的约束，告诉特定的人群，啥事能做？啥事不能做？啥事应该咋做？

作者回复: 回答的非常好，态度认真。

1.感谢支持。

2.区分爬虫这个，完全在于cdn想不想做，因为cdn本质上也是web服务器，如果它认为爬虫影响了服务，就可以反爬虫。如果它决定爬虫也算源站的正常流量，就不反。所以这个问题就是让大家去思考，没有绝对正确的答案。

3.说的很清楚，有自己的思考就是好事。。



👍 8



patsun

2019-06-03

1.CDN在对待浏览器和爬虫时没有差异，因为如果没有验证码或者其他验证方式区分的话，浏览器和爬虫都被视为User Agent（客户代理）

2.Webservice是服务，Web Server是服务器

作者回复:



👍 6

领资料



刘志兵

2019-08-02

老师，服务器只有这么少的几个吗，有一些grpc 服务算服务器吗，finagle, grpc等，还有spring不是也可以提供服务吗

作者回复: 里面说的特指的是“Web服务器”，也就是说专门提供http服务的服务器软件。

其他的像tomcat、netty等虽然也有http服务，但不是专门做这个的，http只是“副业”，功能远不如专业的Apache、Nginx强大。

grpc属于服务开发框架，用的是基于http/2的grpc协议。



👍 5



永钱

2019-06-03

老师把tomcat放在web服务器中比较，说速度慢，不公平呀

作者回复: 抱歉啦，的确，tomcat应该是web容器。



👍 5



lfn

2019-06-04

不是很理解web server和web service的区别。难道我们的服务不用nginx就不能用了么？我自己写个tcp server，根据用户请求调用特定的handler返回数据，那我自己就是个server啊，也是service.老师能不能更清晰地定义下server.习惯了tcp编程的概念，这里的server就显得怪怪的，给人一种router的感觉.

作者回复: 这里首先要理解web，web就是指的http。web server就是提供http服务的server、web service就是运行在http协议上的服务接口规范。

自己写的是tcp server，就不是web server。

service通常是指服务程序，跑在server上，server可以理解成容器、平台。



👍 4



關..

2020-03-09

爬虫作为机器人，承载他的物理容器是什么？那为啥这个就不能作为一个真实的用户去访问数据？

领资料



作者回复: 机器人不一定要有物理实体, 爬虫其实是一个程序, 访问网站抓取内容。

爬虫运行在一台机器上, 由于是自动的, 不代表真实的用户意志, 对于网站来说就没有价值, 你不可能让爬虫去买东西、看广告。



👍 3



欢乐的小马驹

2020-02-24

老师, 你说可以通过User-Agent来区分爬虫, 那他们能假装成浏览器吗? 怎么假装成浏览器呢?

作者回复: 当然可以了, 只要把User-Agent的字符串改成浏览器的就可以了, 因为单从http报文上是区分不出客户端的。

当然, 后台服务器还可以从ip、访问频率等其他方式来判断爬虫, 不会仅依赖User-Agent, 因为它太容易被伪造了。

共 2 条评论 >

👍 3



壹笙 漂泊

2019-06-03

- 1、应该不会有差异, 因为爬虫主要就是无限模仿浏览器行为
- 2、Web Server 是服务器, Web Service 是一种应用服务开发规范

作者回复: ✓



👍 3



不靠谱~

2019-06-03

- 1.不是太清楚, 个人认为不会区别对待, 因为在正常程序应用来说, 看不出是谁发起的请求。
 - 2.web server是软件服务器, 承载应用。
- web service是一种服务方式。

作者回复: ✓



👍 3



多多

2020-11-02

领资料



1. 你觉得 CDN 在对待浏览器和爬虫时会有差异吗？为什么？你怎么理解？
CDN不会管是谁访问自己，本质上它只是一种提供资源加速服务的方式。
如果需要针对爬虫做搜索引擎优化，服务端可以根据User-Agent来判断发起请求的一端是浏览器还是爬虫，对待爬虫可以专门的服务端渲染等。

2. Webservice 与 Web Server 这两个非常相似的词？
一个是服务，一个是服务运行环境

作者回复: 说的很好。

共 2 条评论 >

👍 2



-W.LI-

2019-06-03

第一个有差别，因为有反爬虫技术
第二个:web server 。web服务提供者，web服务器。web应用程序。 web service。。。。
知道了

作者回复: 这个具体还要看cdn的策略，如果配置了反爬虫就会区别对待。

共 3 条评论 >

👍 2



Berry He

2019-06-03

第一个不太清楚，不敢妄加评论。
第二个:web server和web service是两个概念，前者是web服务器，像iis apache nginx这种。web service他只是一个或多个提供web请求响应的api，用来获取或提交更新web server资源的

作者回复: 最后一句话不太准确，web service是应用服务，它的客户端不一定是web service。



👍 2



Inner peace

2019-06-13

1. 不会区别对待，当爬虫的浏览器都使用一样的useragent时，cdn并不能识别
2. Web server是web服务器，web service是一种web服务协议

作者回复: √

领资料





1



鸟人

2019-06-04

你觉得 CDN 在对待浏览器和爬虫时会有差异吗？为什么？
不管是否反爬虫 应该都没区别 爬虫本质不就是模拟浏览器么

作者回复: 最后一句话不太准确，爬虫应该是user agent的一种，不一定非要模拟浏览器。



1

领资料

