

08 | 键入网址再按下回车，后面究竟发生了什么？

2019-06-14 Chrono

《透视HTTP协议》

课程介绍 >



讲述：Chrono

时长 11:30 大小 13.18M



经过上一讲的学习，你是否已经在自己的电脑上搭建好了“最小化”的 HTTP 实验环境呢？

我相信你的答案一定是“Yes”，那么，让我们立刻开始“螺蛳壳里做道场”，在这个实验环境里看一下 HTTP 协议工作的全过程。

使用 IP 地址访问 Web 服务器

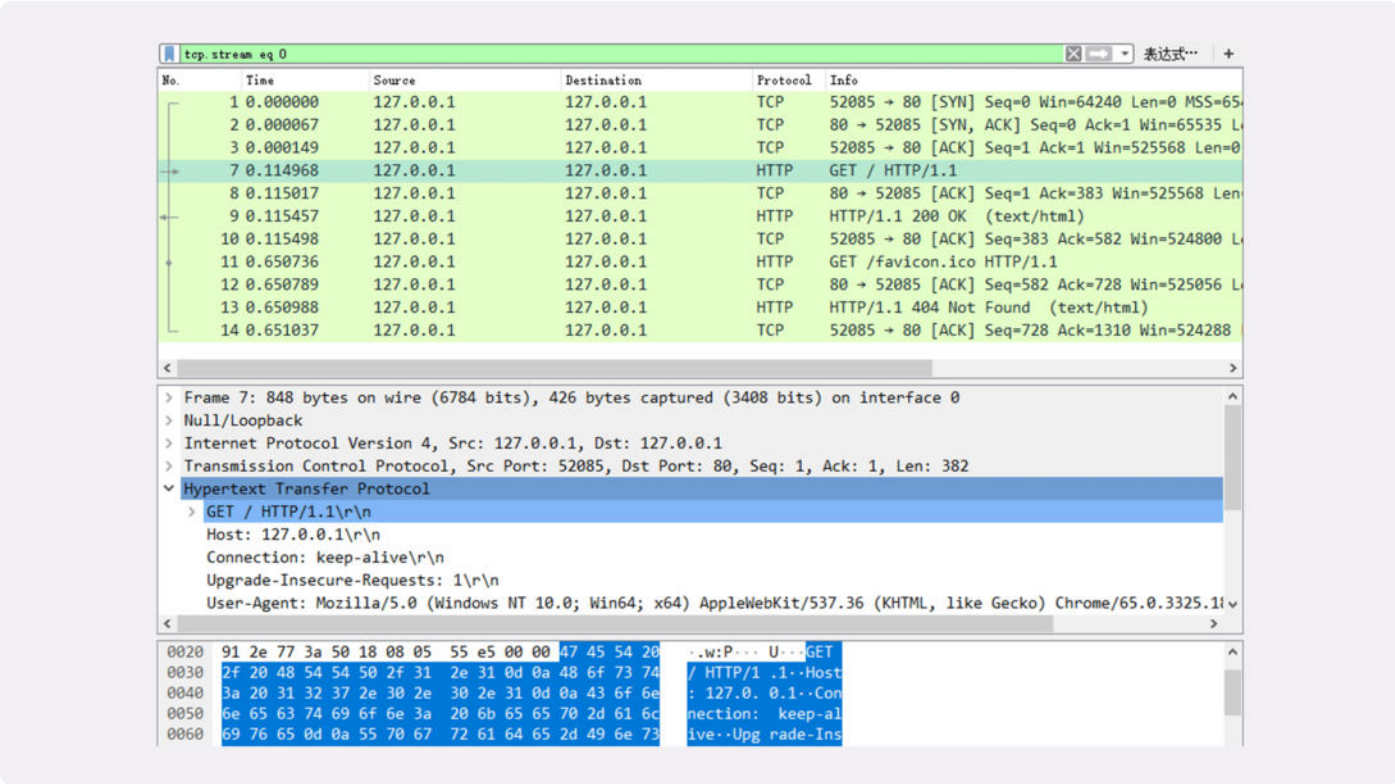
首先我们运行 www 目录下的“start”批处理程序，启动本机的 OpenResty 服务器，启动后可以用“list”批处理确认服务是否正常运行。

领资料

然后我们打开 Wireshark，选择“HTTP TCP port(80)”过滤器，再鼠标双击“Npcap loopback Adapter”，开始抓取本机 127.0.0.1 地址上的网络数据。



第三步，在 Chrome 浏览器的地址栏里输入“<http://127.0.0.1/>”，再按下回车键，等欢迎页面显示出来后 Wireshark 里就会有捕获的数据包，如下图所示。



如果你还没有搭好实验环境，或者捕获与本文里的不一致也没关系。我把这次捕获的数据存成了 pcap 包，文件名是“08-1”，放到了 GitHub 上，你可以下载到本地后再用 Wireshark 打开，完全精确“重放”刚才的 HTTP 传输过程。

抓包分析

在 Wireshark 里你可以看到，这次一共抓到了 11 个包（这里用了滤包功能，滤掉了 3 个包，原本是 14 个包），耗时 0.65 秒，下面我们就来一起分析一下"键入网址按下回车"后数据传输的全过程。

通过前面“破冰篇”的讲解，你应该知道 HTTP 协议是运行在 TCP/IP 基础上的，依靠 TCP/IP 协议来实现数据的可靠传输。所以浏览器要用 HTTP 协议收发数据，首先要做的就是建立 TCP 连接。

因为我们在地址栏里直接输入了 IP 地址“127.0.0.1”，而 Web 服务器的默认端口是 80，所以浏览器就要依照 TCP 协议的规范，使用“三次握手”建立与 Web 服务器的连接。



对应到 Wireshark 里，就是最开始的三个抓包，浏览器使用的端口是 52085，服务器使用的端口是 80，经过 SYN、SYN/ACK、ACK 的三个包之后，浏览器与服务器的 TCP 连接就建立起来了。

有了可靠的 TCP 连接通道后，HTTP 协议就可以开始工作了。于是，浏览器按照 HTTP 协议规定的格式，通过 TCP 发送了一个“GET / HTTP/1.1”请求报文，也就是 Wireshark 里的第四个包。至于包的内容具体是什么现在先不用管，我们下一讲再说。

随后，Web 服务器回复了第五个包，在 TCP 协议层面确认：“刚才的报文我已经收到了”，不过这个 TCP 包 HTTP 协议是看不见的。

Web 服务器收到报文后在内部就要处理这个请求。同样也是依据 HTTP 协议的规定，解析报文，看看浏览器发送这个请求想要干什么。

它一看，原来是要求获取根目录下的默认文件，好吧，那我就从磁盘上把那个文件全读出来，再拼成符合 HTTP 格式的报文，发回去吧。这就是 Wireshark 里的第六个包“HTTP/1.1 200 OK”，底层走的还是 TCP 协议。

同样的，浏览器也要给服务器回复一个 TCP 的 ACK 确认，“你的响应报文收到了，多谢”，即第七个包。

这时浏览器就收到了响应数据，但里面是什么呢？所以也要解析报文。一看，服务器给我的是个 HTML 文件，好，那我就调用排版引擎、JavaScript 引擎等等处理一下，然后在浏览器窗口里展现出了欢迎页面。

这之后还有两个来回，共四个包，重复了相同的步骤。这是浏览器自动请求了作为网站图标的“favicon.ico”文件，与我们输入的网址无关。但因为我们的实验环境没有这个文件，所以服务器在硬盘上找不到，返回了一个“404 Not Found”。

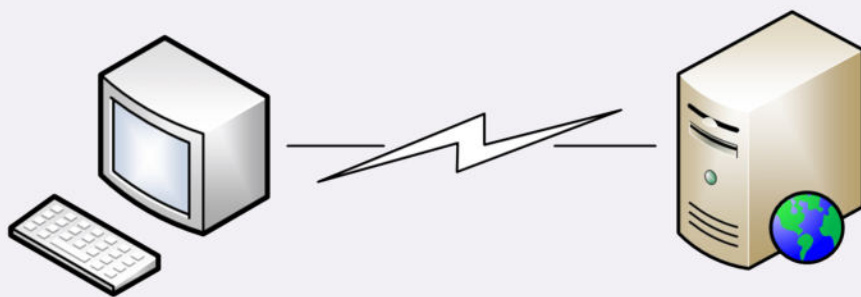
至此，“键入网址再按下回车”的全过程就结束了。

我为这个过程画了一个交互图，你可以对照着看一下。不过要提醒你，图里 TCP 关闭连接的“四次挥手”在抓包里没有出现，这是因为 HTTP/1.1 长连接特性，默认不会立即关闭连接。

领资料



127.0.0.1

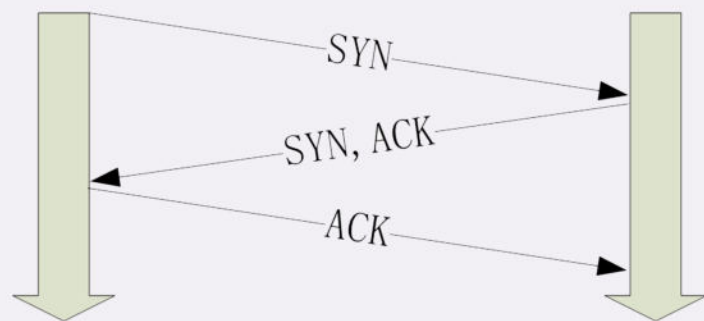


浏览器

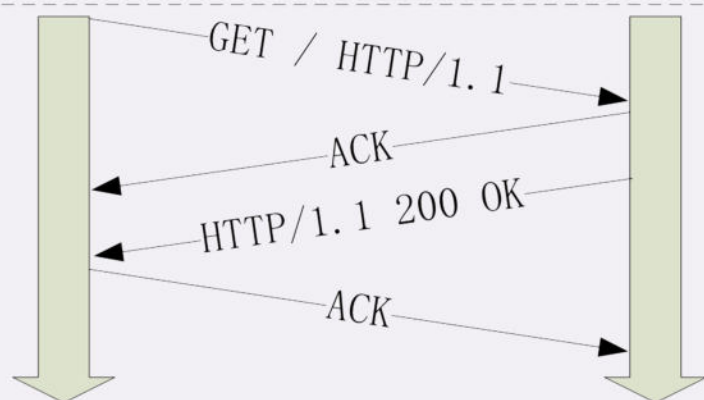
127.0.0.1:5xxx

服务器

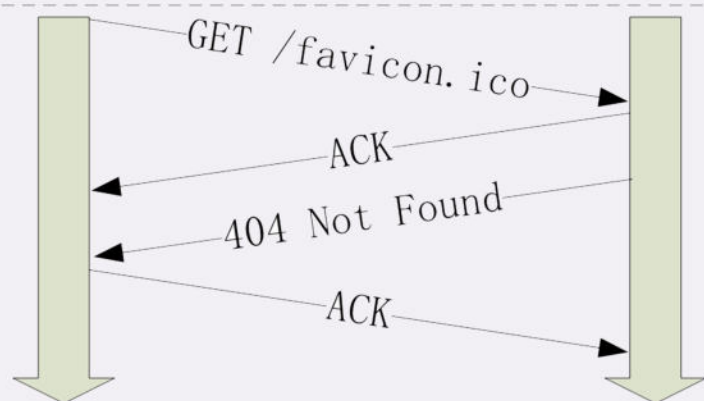
127.0.0.1:80



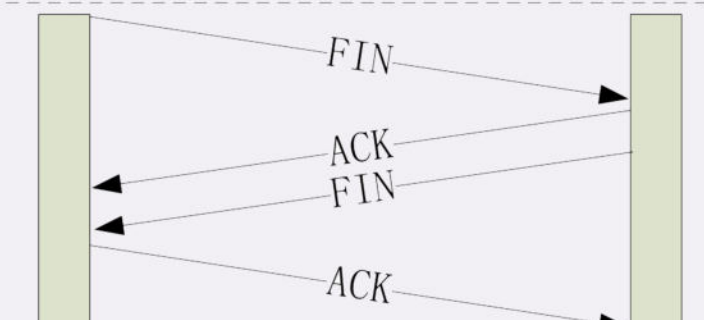
TCP三次握手
建立连接



HTTP传输
HTML页面




HTTP传输
图标文件



TCP四次挥手
关闭连接

领资料





再简要叙述一下这次最简单的浏览器 HTTP 请求过程：

1. 浏览器从地址栏的输入中获得服务器的 IP 地址和端口号；
2. 浏览器用 TCP 的三次握手与服务器建立连接；
3. 浏览器向服务器发送拼好的报文；
4. 服务器收到报文后处理请求，同样拼好报文再发给浏览器；
5. 浏览器解析报文，渲染输出页面。

使用域名访问 Web 服务器

刚才我们是在浏览器地址栏里直接输入 IP 地址，但绝大多数情况下，我们是不知道服务器 IP 地址的，使用的是域名，那么改用域名后这个过程会有什么不同吗？

还是实际动手试一下吧，把地址栏的输入改成“<http://www.chrono.com>”，重复 Wireshark 抓包过程，你会发现，好像没有什么不同，浏览器上同样显示出了欢迎界面，抓到的包也同样是 11 个：先是三次握手，然后是两次 HTTP 传输。

这里就出现了一个问题：浏览器是如何从网址里知道“www.chrono.com”的 IP 地址就是“127.0.0.1”的呢？

还记得我们之前讲过的 DNS 知识吗？浏览器看到了网址里的“www.chrono.com”，发现它不是数字形式的 IP 地址，那就肯定是域名了，于是就会发起域名解析动作，通过访问一系列的域名解析服务器，试图把这个域名翻译成 TCP/IP 协议里的 IP 地址。

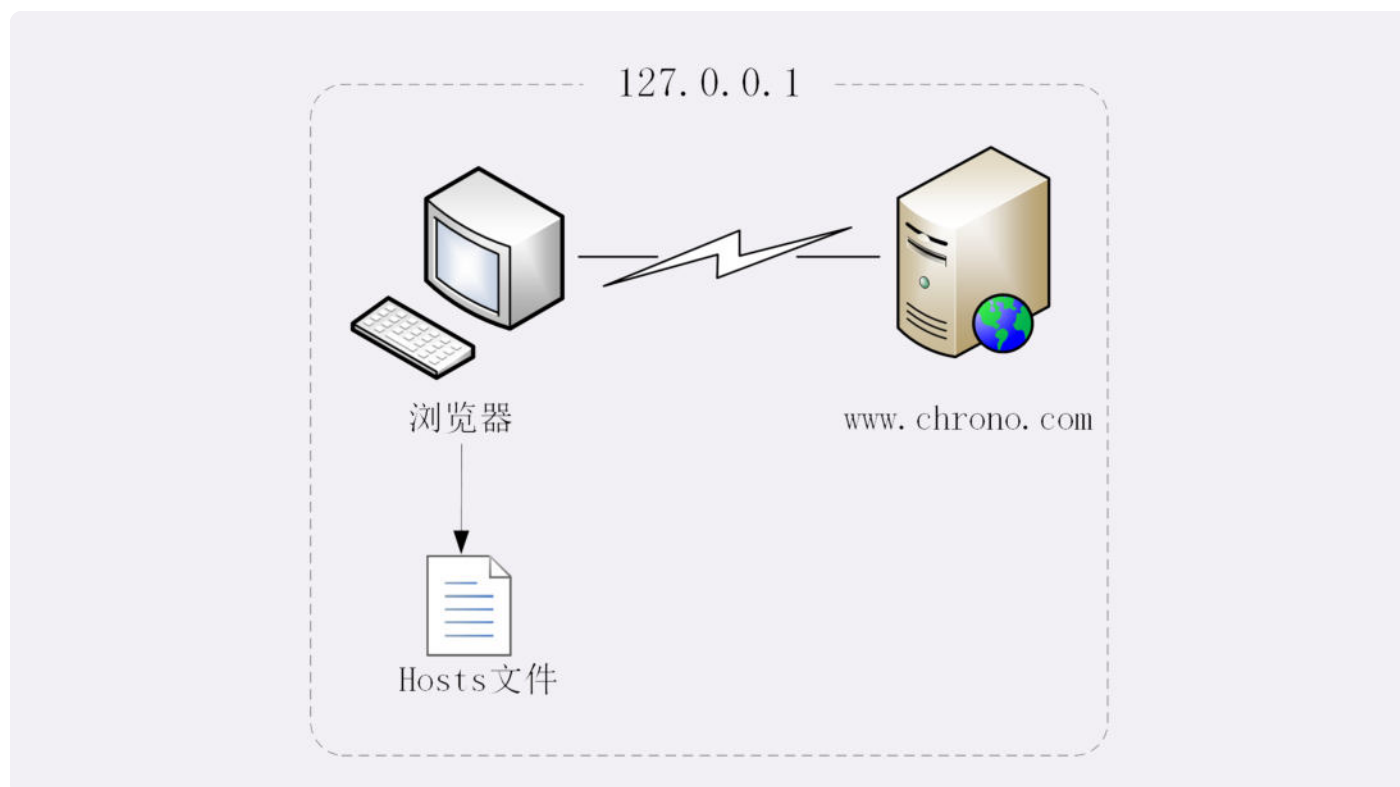
不过因为域名解析的全过程实在是太复杂了，如果每一个域名都要大费周折地去网上查一下，那我们上网肯定会慢得受不了。

所以，在域名解析的过程中会有多级的缓存，浏览器首先看一下自己的缓存里有没有，如果没有就向操作系统的缓存要，还没有就检查本机域名解析文件 hosts，也就是上一讲中我们修改的“C:\WINDOWS\system32\drivers\etc\hosts”。



刚好，里面有一行映射关系“127.0.0.1 www.chrono.com”，于是浏览器就知道了域名对应的 IP 地址，就可以愉快地建立 TCP 连接发送 HTTP 请求了。

我把这个过程也画出了一张图，但省略了 TCP/IP 协议的交互部分，里面的浏览器多出了一个访问 hosts 文件的动作，也就是本机的 DNS 解析。



真实的网络世界

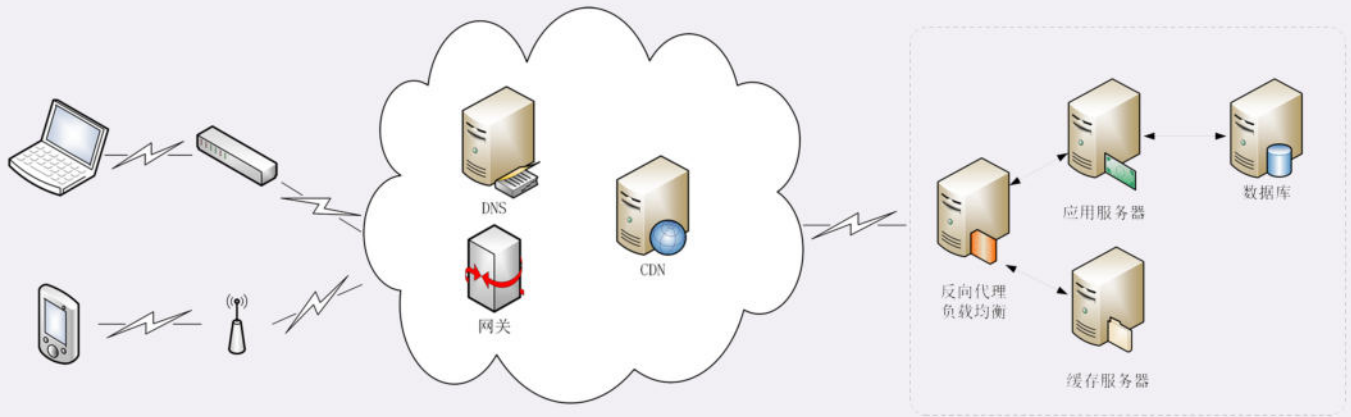
通过上面两个在“最小化”环境里的实验，你是否已经对 HTTP 协议的工作流程有了基本的认识呢？

第一个实验是最简单的场景，只有两个角色：浏览器和服务器，浏览器可以直接用 IP 地址找到服务器，两者直接建立 TCP 连接后发送 HTTP 报文通信。

第二个实验在浏览器和服务器之外增加了一个 DNS 的角色，浏览器不知道服务器的 IP 地址，所以必须要借助 DNS 的域名解析功能得到服务器的 IP 地址，然后才能与服务器通信。

真实的互联网世界要比这两个场景要复杂的多，我利用下面的这张图来做一个详细的说明。





如果你用的是电脑台式机，那么你可能会使用带水晶头的双绞线连上网口，由交换机接入固定网络。如果你用的是手机、平板电脑，那么你可能会通过蜂窝网络、WiFi，由电信基站、无线热点接入移动网络。

接入网络的同时，网络运行商会给你的设备分配一个 IP 地址，这个地址可能是静态分配的，也可能是动态分配的。静态 IP 就始终不变，而动态 IP 可能你下次上网就变了。

假设你要访问的是 Apple 网站，显然你是不知道它的真实 IP 地址的，在浏览器里只能使用域名“www.apple.com”访问，那么接下来要做的必然是域名解析。这就要用 DNS 协议开始从操作系统、本地 DNS、根 DNS、顶级 DNS、权威 DNS 的层层解析，当然这中间有缓存，可能不会费太多时间就能拿到结果。

别忘了互联网上还有另外一个重要的角色 CDN，它也会在 DNS 的解析过程中“插上一脚”。DNS 解析可能会给出 CDN 服务器的 IP 地址，这样你拿到的就会是 CDN 服务器而不是目标网站的实际地址。

因为 CDN 会缓存网站的大部分资源，比如图片、CSS 样式表，所以有的 HTTP 请求就不需要再发到 Apple，CDN 就可以直接响应你的请求，把数据发给你。

由 PHP、Java 等后台服务动态生成的页面属于“动态资源”，CDN 无法缓存，只能从目标网站获取。于是你发出的 HTTP 请求就要开始在互联网上的“漫长跋涉”，经过无数的路由器、

领资料



网关、代理，最后到达目的地。

目标网站的服务器对外表现的是一个 IP 地址，但为了能够扛住高并发，在内部也是一套复杂的架构。通常在入口是负载均衡设备，例如四层的 LVS 或者七层的 Nginx，在后面是许多的服务器，构成一个更强更稳定的集群。

负载均衡设备会先访问系统里的缓存服务器，通常有 memory 级缓存 Redis 和 disk 级缓存 Varnish，它们的作用与 CDN 类似，不过是工作在内部网络里，把最频繁访问的数据缓存几秒钟或几分钟，减轻后端应用服务器的压力。

如果缓存服务器里也没有，那么负载均衡设备就要把请求转发给应用服务器了。这里就是各种开发框架大显神通的地方了，例如 Java 的 Tomcat/Netty/Jetty，Python 的 Django，还有 PHP、Node.js、Golang 等等。它们又会再访问后面的 MySQL、PostgreSQL、MongoDB 等数据库服务，实现用户登录、商品查询、购物下单、扣款支付等业务操作，然后把执行的结果返回给负载均衡设备，同时也可能给缓存服务器里也放一份。

应用服务器的输出到了负载均衡设备这里，请求的处理就算是完成了，就要按照原路再走回去，还是要经过许多的路由器、网关、代理。如果这个资源允许缓存，那么经过 CDN 的时候它也会做缓存，这样下次同样的请求就不会到达源站了。

最后网站的响应数据回到了你的设备，它可能是 HTML、JSON、图片或者其他格式的数据，需要由浏览器解析处理才能显示出来，如果数据里面还有超链接，指向别的资源，那么就又要重走一遍整个流程，直到所有的资源都下载完。

小结

今天我们在本机环境里做了两个简单的实验，学习了 HTTP 协议请求 – 应答的全过程，在这里做一个小结。

1. HTTP 协议基于底层的 TCP/IP 协议，所以必须要用 IP 地址建立连接；
2. 如果不知道 IP 地址，就要用 DNS 协议去解析得到 IP 地址，否则就会连接失败；
3. 建立 TCP 连接后会顺序收发数据，请求方和应答方都必须依据 HTTP 规范构建和解析报文；
4. 为了减少响应时间，整个过程中的每一个环节都会有缓存，能够实现“短路”操作；



5. 虽然现实中的 HTTP 传输过程非常复杂，但理论上仍然可以简化成实验里的“两点”模型。

课下作业

1. 你能试着解释一下在浏览器里点击页面链接后发生了哪些事情吗？
2. 这一节课里讲的都是正常的请求处理流程，如果是一个不存在的域名，那么浏览器的工作流程会是怎样的呢？

欢迎你把自己的答案写在留言区，与我和其他同学一起讨论。如果你觉得有所收获，也欢迎把文章分享给你的朋友。

领资料






== 课外小贴士 ==

- 01 除了 80 端口，HTTP 协议还经常使用 8000 和 8080。
- 02 因为 Chrome 浏览器会缓存之前访问过的网站，所以当你再次访问“127.0.0.1”的时候它可能会直接从本地缓存而不是服务器获取数据，这样就无法用 Wireshark 捕获网络流量，解决办法是在 Chrome 的开发者工具或者设置里清除相关的缓存。
- 03 现代浏览器通常都会自动且秘密地发送 favicon.ico 请求。



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



上一篇 07 | 自己动手，搭建HTTP实验环境

下一篇 09 | HTTP报文是什么样子的？

学习推荐

JVM + NIO + Spring

各大厂面试题及知识点详解

限时免费



精选留言 (103)

写留言



-W.LI-

2019-06-14

浏览器判断是不是ip地址，不是就进行域名解析，依次通过浏览器缓存，系统缓存，host文件，还是没找到的请求DNS服务器获取IP解析(解析失败的浏览器尝试换别的DNS服务器，最终失败的进入错误页面)，有可能获取到CDN服务器IP地址，访问CDN时先看是否缓存了，缓存了响应用户，无法缓存，缓存失效或者无缓存，回源到服务器。经过防火墙外网网管路由到nginx接入层。ng缓存中存在的直接放回，不存在的负载到web服务器。web服务器接受到请求后处理，路径不存在404。存在的返回结果(服务器中也会有redis,ehcache(堆内外缓存)，disk等缓存策略)。原路返回，CDN加入缓存响应用户。

作者回复: 说的非常详细。

领资料



**郭凯强**

2019-06-14

作业:

1. 浏览器判断这个链接是要在当前页面打开还是新开标签页，然后走一遍本文中的访问过程：拿到ip地址和端口号，建立tcp/ip链接，发送请求报文，接收服务器返回并渲染。
2. 先查浏览器缓存，然后是系统缓存->hosts文件->局域网域名服务器->广域网域名服务器->顶级域名服务器->根域名服务器。这个时间通常要很久，最终找不到以后，返回一个报错页面，chrome是ERR_CONNECTION_ABORTED

作者回复: 回答的比较全面。

这里面还有个长连接的问题，后面会讲，如果连接还是本站就不会有建连过程，直接用已有的连接发请求。

共 3 条评论 >

24

**温木**

2019-08-06

老师，我有一个问题请教。

DNS域名解析不需要发请求，建立连接吗？本地缓存的dns除外。

比如我第一次访问一个域名abc.com,那这第一次不是需要从dns服务器上拿真正的IP吗，去拿IP的这个过程不是应该也是一个请求吗？这个请求又是什么请求呢？

作者回复: dns请求是专门的dns协议，使用udp发送，因为是udp所以不需要建立连接。



22

**极客时间**

2019-06-14

老师 我有个疑问，第四个包到第六个包，为什么又进行了一次tcp连接呢，而且这个端口号是52086，这个是浏览器的特性吗，仔细比对文章发现问题啊

作者回复: 因为http/1连接传输效率低，所以浏览器一般会对同一个域名发起多个连接提高效率，这个52086就是开的第二个连接，但在抓包中只是打开了，还没有传输。

到后面讲长连接的时候你就会明白了。

共 4 条评论 >

18

领资料





我有几个小疑问没搞明白，万望老师解答，在进行DNS解析的时候，操作系统和本地DNS是如何处理的呢？

我的理解是本地系统有可能有缓存，DNS解析前先查看本地有没有缓存，如果没有缓存，再进行本地DNS解析，本地DNS解析就是查找系统里面的hosts文件的对应关系。不知道这里理解的对不对。

还有一个疑问。

什么是权威DNS呢，我一般是在万网购买域名，然后用A记录解析到我的服务器，这个A记录提交到哪里保存了呢，这里的万网扮演的是什么角色呢？它和权威DNS有关系吗？

上次我提到了一个问题，就是域名和ip的对应关系，没接触这个课程以前，我的理解是一个域名只能解析到一个ip地址，但是一个ip地址可以绑定多个域名，就像一个人只有一个身份证号码，但是可以有多个名字，但是我在用ping命令 ping ' baidu.com' 时，发现 可以返回不同的ip，结合本课程前面的文章，我理解是百度自己的服务器本质是一台DNS服务器，用DNS做了负载均衡，当我访问baidu.com时，域名解析过程中，有一个环节是到达了百度的DNS服务器，然后DNS服务器根据负载均衡操作，再将我的请求转发给目标服务器。不知道理解的对不对，或者哪里有偏差。

作者回复：本地dns你的理解是正确的。

万网是个域名注册的代理机构，最终域名还是要由dns系统来解析。

百度的理解基本正确，在真正服务器前面是dns负载均衡。

共 4 条评论 >

👍 12



肥low

2019-06-17

1、如果域名不是ip，需要走域名解析成ip的逻辑，优先级顺序为：1 浏览器缓存 > 2 本地hosts > 3 系统缓存 > 4 根域名 > 5 顶级dns服务器(如 com) > 6 二级dns服务器(baidu.com) > 7 三级dns服务器(www.baidu.com)，如果客户端指向的dns服务器为非官方的如 8.8.8.8，那在第4步之前可能还有一层cache,当然最后解析的ip有可能是cdn的，如果cdn失效了就直接穿透到源ip，当然这个服务器这一部分可能做了四层负载均衡的设置，所以有可能每次获取的服务器ip都不一样，也有可能到了服务器ngx层做了七层转发，所以虽然获得的ip一样，但是内部可能转发给了很多内网服务器

2、通过中间各种路由器的转发，找到了最终服务器，进行tcp三次握手，数据请求，请求分两种一种是uri请求，一种是浏览器咸吃萝卜淡操心的请求网站图标ico的资源请求，然后服务端收到请求后进行请求分析，最终返回http报文，再通过tcp这个连接隧道返回给用户端，用户端收到后再告诉服务端已经收到结果的信号(ack)，然后客户端有一套解析规则，如果是ht

领资料



ml, 可能还有额外的外部连接请求, 是跟刚才的请求流程是同理的(假设是http1.1), 只不过没有了tcp三次握手的过程, 最终用户看到了百度的搜索页面。当然如果dns没解析成功, 浏览器直接就报错了, 不会继续请求接下来的资源

作者回复: 非常详细, 赞!!

共 5 条评论 >

👍 10



四月的紫色花

2019-07-30

1.你能试着解释一下在浏览器里点击页面链接后发生了什么事情吗?

浏览器点击页面请求后, 正常网络中都是域名, 那么浏览器会先用DNS解析一下, 拿到服务器的ip和端口, 去请求服务器前会先找一下缓存, 浏览器自己的缓存-操作系统缓存-本地缓存(Hosts), 都没有的话就会到根域名服务器-顶级-权威, 当然中间可能有类似CDN这样的代理, 那它就可以取CDN中的服务器地址, 总的来说, 其实就是个“走近道”的过程, 就近原则, 在DNS不错的情况下, 先从离自己近的查起, 再一级一级往下。

2.这一节课里讲的都是正常的请求处理流程, 如果是一个不存在的域名, 那么浏览器的工作流程会是怎么样的呢?

如果是一个不存在的域名, 那浏览器还是会从DNS那解析一下, 发现, 自己, 操作系统, 本地的缓存都没有, CDN里也没有, 根域名, 顶级域名, 权威域名, 非权威域名里都没有, 那它就放弃了, 不会建立链接, 返回错误码, 可能是4xx类的客户端请求错误。

作者回复: 回答的很认真, 鼓励一下。

第二个问题后面有误, 因为dns解析失败, 根本没有进入http处理流程, 所以不会有4xx之类的错误, 而是dns解析错误信息。

共 2 条评论 >

👍 7



陈1016

2019-06-27

第一个问题的回答: 浏览器缓存、系统缓存、hosts文件、野生DNS服务器(本地DNS服务器)、根DNS、顶级DNS、权威DNS、本地(附近)CDN、源站。

作者回复: √

共 2 条评论 >

👍 7

领资料



Maske

2020-06-08

1.如果链接地址是域名开头的, 浏览器会开始DNS解析动作。解析优先级依次为: 浏览器缓存

> 操作系统缓存 > 本机hosts文件 > “野生DNS服务器” > 核心DNS服务器（根级DNS > 顶级DNS > 权威DNS）；将域名解析为正确的ip地址之后，通过三次握手与服务器建立tcp/ip连接；浏览器发送请求报文，服务器接收并处理请求，返回响应报文，浏览器开始解析html文档，在这过程中又会发起一些http请求，进行图片、css、js等静态资源的获取，以及ajax请求获取json数据。同时，浏览器相关引擎开始绘制dom视图，执行js脚本，完成页面的初始化直到所有代码执行完毕。

2.如1中所说DNS解析顺序，当请求DNS服务器进行域名解析时，发现没有找到对应的ip，会导致解析失败，无法建立tcp/ip链接，导致浏览器建立连接时间过长，最终建立连接失败，浏览器停止建立连接动作。

作者回复: 说的非常好!



👍 4



keep it simple

2019-11-27

老师，学习这一章萌生出几个问题：

- 1.如果在TCP连接保持的情况下某一方突然断电了，没有机会进行TCP 四次挥手，会出现什么情况呢？
- 2.如果不主动关浏览器，TCP连接好像一直存在着，会有超时时间吗？中间是否会保活？
- 3.若server端负载较高，当它收到client的SYN包时，是否要过一段时间才会回应SYN,ACK？

作者回复:

1.tcp收发数据有超时机制，超时没有响应就会断开连接，当然这个就不是正常的结束连接。

2.tcp收发有超时，连接本身没有超时机制，http使用keepalive在tcp上实现了连接保活。

3.tcp建连是在操作系统内核里实现的，有一个处理队列，如果并发的请求太多，就会排队等待。

4.这些问题涉及的都是tcp比较底层的细节，我也不能很好的解释清楚，建议再去参考其他的资料，sorry。



👍 3

领资料



徐徐

2019-08-17

你好，罗老师

我在本地测试了一下，结果有点不解

- 1、浏览器上访问了一次127.0.0.1，发起了两次：三次握手，四次握手；但没有访问/favicon.ico；对应端口分别是52181->80、52182->80。



2、52181在四次挥手是服务端先发起了：[FIN,ACK]，客户端：[ACK],[FIN,ACK]，服务端：[ACK]，和你画的四次挥手顺序不对，52182和52181四次挥手顺序保持一致。

作者回复：

1.这是浏览器建立了两个并发连接，没有访问favicon也是正常的，跟浏览器有关。

2.这个应该是服务器主动关闭连接。

3.课程里的示例是挑选了一个最典型的场景，并不是所有的请求响应都会按照这个来。



👍 3



乐雨

2019-08-16

操作系统缓存是指什么？我理解就是hosts文件，为什么dns解析时分成了两步？

作者回复: hosts文件相当于是一个简易的dns解析器（KV格式），而操作系统缓存则是在内存里，访问缓存要比访问磁盘快的多。

所以解析dns都要先找缓存，没有才去访问解析器（hosts、dns服务器等）。



👍 2



张德

2019-08-01

我记得有一年北大计算机专业的考研就有这一个题 😊

作者回复: 看来学http不仅对于参加工作的同学有用的，对于还在象牙塔里的同学也有用啊。

共 3 条评论 >

👍 2



隰有荷

2019-06-15

老师，文中在讲解请求Apple网站的例子时，说到：

"这就要用 DNS 协议开始从操作系统、本地 DNS、根 DNS、顶级 DNS、权威 DNS 的层层解析"。

而我看前几天的内容总结的是，请求会在进入网络后先到达非权威DNS、权威DNS、顶级...、最后才是到达根DNS去解析，那么本文怎么是先从根DNS开始的呢？

作者回复: 今天的这讲是简化的说法，没有那么精确，完整的dns解析以第6讲为准。

领资料



另外，dns解析通常是先到非权威dns，然后是根dns->顶级dns->权威dns，你可以再回顾一下。



2



学不动了

2021-05-26

在抓包的工程中，wireshark出现：

- 1.红色的RST/ACK的包，
- 2.三次握手之后立马发送了[TCP Window Update]

对于问题1，查到的资料说是RST是复位标识，异常关闭连接

问题2，就没有找到明确的答案，麻烦老师解答下 为什么三次握手之后又发送了一个[TCP Window Update]？

作者回复：这个就是tcp层的知识了，根据当前网络的状况，调整数据收发的速度。



1



Geek_Maggie

2021-03-08

作业1： 在浏览器里点击页面链接后发生了什么事情吗？

1. 页面链接是一个个URL，点击链接相当于在浏览器输入目标服务器的IP及端口后按下回车（有时候是域名及端口。端口http默认是80，https默认是443，这两个端口默认不写。）
2. 此时浏览器作为user agent发起http请求，这时候会先和目标服务器通过“三次握手”建立TCP连接
3. 建立连接后，服务器接收到浏览器的报文，解析浏览器发起的请求要访问什么资源。如果报文解析正确，知晓浏览器要访问的资源，则返回http返回码和所请求内容；如其他情况，返回其他http状态码进行告知。
4. 浏览器接收返回的信息，使用http模板引擎、JS、CSS等引擎将静态网页和资源渲染出来。
5. 如果是链接是带域名的，在建立TCP连接前，会进行域名解析的步骤：先访问本地缓存（浏览器缓存，然后是操作系统缓存、host文件）找不到则再逐级DNS（野生DNS服务器-根-顶级-权威）查找。域名如果是CDN的域名，可能先访问CDN的缓存资源，找不到再去源站。

作业2. 如果是一个不存在的域名，那么浏览器的工作流程会是怎么样的呢？

不存在的域名也会去解析，先执行域名解析的步骤：先访问本地缓存（浏览器缓存，然后是操作系统缓存、host文件）找不到则再逐级DNS（野生DNS服务器-根-顶级-权威），都执行完还是找不到就报错。

作者回复：说的很好。

领资料





1

**nora**

2019-11-25

有个疑问是，如果DNS解析得到的IP已经失效，并不是当前最新的IP，会发生什么呢？

作者回复：那就会在tcp层次发生连接失败，或者连接到了错误的其他网站。

老师，接这个问题，提出两个疑问

1. 发生链接错误时，会重新进行DNS解析嘛？
2. 假如连接到了错误的IP，那么什么时候会更新成为正确的IP呢？或者有什么办法可以更正DNS的解析错误

作者回复：

1.不会，因为一次请求已经完成了，这种错误计算机不会认为是dns错误，因为处理流程都是正确的，只是得到的ip地址数据错误。

2.可以手动改hosts，强行指定正确的ip，或者等待dns服务恢复正常，后者在dns被污染的时候就很难说了。

共 2 条评论 >



1

**错夕**

2019-09-29

关于浏览器回车发生了什么我觉得应该关注的是

- 1.是否一定会进行DNS解析如果有缓存这个时候该怎么办
- 2.是否一定会进行TCP链接
- 3.收到服务器相应报文一定会解析dom进行渲染吗？如果返回304呢
- 4.TCP链接何时关闭

作者回复：

1.如果输入的不是ip地址，那么必然会做dns解析，否则没有ip就无法访问。而dns解析有可能会直接从缓存里取，不一定会走到解析服务器。

2.必须要有tcp连接，因为互联网和http就建立在tcp/ip之上。

3.这个与客户端有关，如果用的是curl这样的命令行就没有渲染。

4.短连接会立即关闭连接，而长连接会保持较长的时间，直到客户端或者服务器某一方主动断开连接。



1

领资料



qiezeitx

2019-09-24

作业

1、浏览器判断是否输入的是ip地址，不是的话就进行DNS解析：浏览器缓存->操作系统缓存（看到老师后面的解释是缓存）->host文件->根DNS->顶级DNS->权威DNS；如果是就开始访问的过程：拿到ip和端口号->3次握手建立TCP连接->发送请求->服务端返回结果->浏览器解析渲染。

这里有几个课外小贴士的实验和疑问：

- 1、ip要么直接输入要么通过DNS解析拿到，但端口号呢？默认端口号不行的时候，会尝试其他候选么？
- 2、经试验，确实有缓存，刷新访问会返回304 not Modified
- 3、经试验，没抓到favicon.ico的流程，chrome可能秘密发送了🐼

2、同留言：属于dns解析错误

作者回复：

1.如果uri里没有端口号，默认用80或者8080、443，但如果在这两个端口上不提供http服务，那浏览器就猜不出来，只能用户手动输入。比如后面安全篇里，就用了442等端口，必须手动输入。

2.实验出真知。

3.有的时候浏览器也不会发favicon请求，可能已经缓存了，这跟浏览器的工作流程有关，不必太在意。



1



Cris

2019-08-13

老师图里的SYN和ACK是什么意思？

作者回复：是tcp里的握手包。



1

领资料

