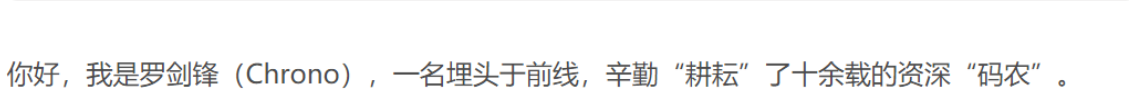
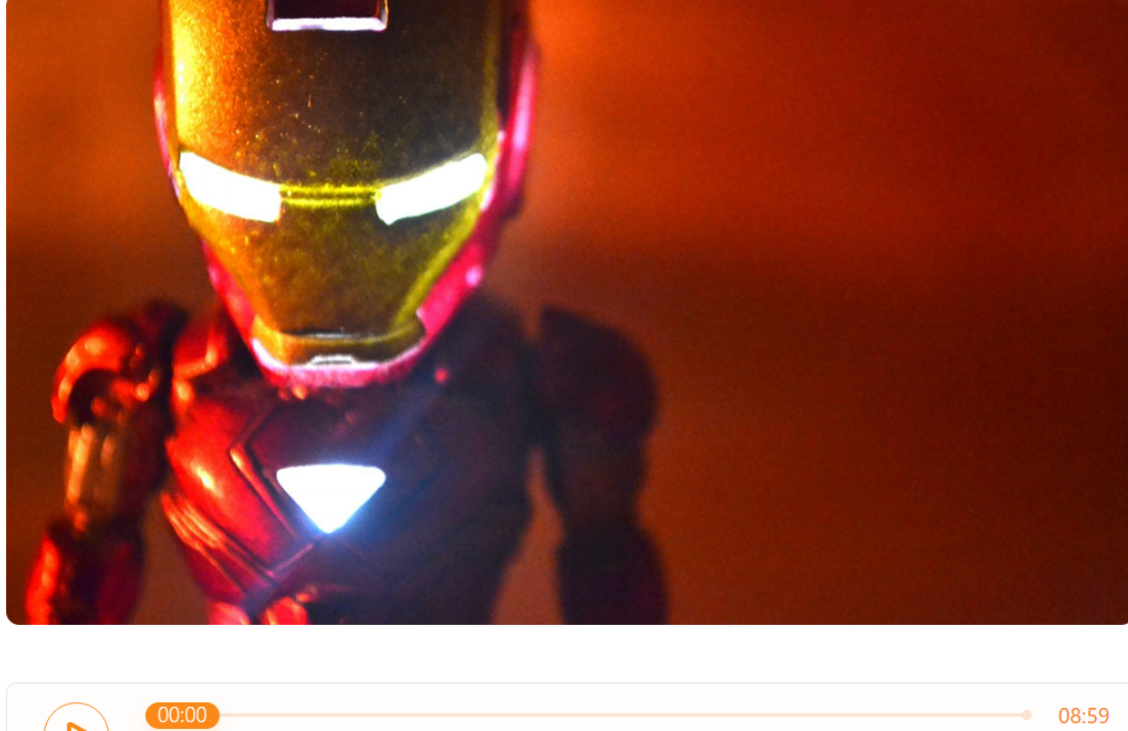


开篇词 | To Be a HTTP Hero

Chrono 2019-05-29



你好，我是罗剑锋（Chrono），一名埋头于前线，辛勤“耕耘”了十余载的资深“码农”。

工作的这十多年来，我开发过智能 IC 卡，也倒腾过商用密码机；做过政务项目，也做过商务搜索；写过网游核心引擎，也写过 CDN 存储系统；在 Windows 上用 C/C++ 做客户端，在 AIX、Linux 上用 Java、PHP 写后台服务.....现在则是专注于“魔改”Nginx，深度定制实现网络协议的分析与检测。

当极客时间的编辑联系我，要我写 HTTP 专栏的时候，我的第一反应是：“HTTP 协议好简单的，有这个必要吗？”

你可能也会有同样的想法：“HTTP 不就是请求 / 响应、GET/POST、Header/Body 吗？网络上的资料一抓一大把，有什么问题搜一下就是了。”

不瞒你说，我当时就是这么想的，在之前的工作中也是一直这么做的，而且一直“感觉良好”，觉得 HTTP 就是这个样子，没有什么特别的地方，没有什么值得讲的。

但在编辑的一再坚持下，我“勉为其难”接下了这个任务。然后做了一个小范围的“调查”，问一些周围的同事，各个领域的都有，比如产品、开发、运维、测试、前端、后端、手机端.....想看看他们有什么意见。

出乎我的意料，他们无一例外都对这个“HTTP 专栏”有很强烈的需求，想好好“补补课”，系统地学习了解 HTTP，这其中甚至还包括有七、八年（甚至更多）工作经验的老手。

这不禁让我陷入了思考，为什么如此“简单”的协议却还有这么多的人想要学呢？

我想，一个原因可能是 HTTP 协议“太常见”了。就像现实中的水和空气一样，如此重要却又如此普遍，普遍到我们几乎忽视了它的存在。真的很像那句俗语所说：“鱼总是最后看见水的”，但水对鱼的生存却又是至关重要。

我认真回忆了一下这些年的工作经历，这才发现 HTTP 只是表面上显得简单，而底层的运行机制、工作原理绝不简单，可以说是非常地复杂。只是我们平常总是“KPI 优先”，网上抓到一个解决方法用过就完事了，没有去深究里面的要点和细节。

下面的几个场景，都是我周围同事的实际感受，你是否也在工作中遇到过这样的困惑呢？你能把它们都解释清楚吗？

- 用 Nginx 搭建 Web 服务器，照着网上的文章配好了，但里面那么多的指令，什么 keepalive、rewrite、proxy_pass 都是怎么回事？为什么要这么配置？
- 用 Python 写爬虫，URI、URL “傻傻分不清”，有时里面还会加一些奇怪的字符，怎么处理才好？
- 都说 HTTP 缓存很有用，可以大幅度提升系统性能，可它是怎么做到的？又应该用在何时何地？
- HTTP 和 HTTPS 是什么关系？还经常听说有 SSL/TLS/SNI/OCSP/ALPN.....这么多稀奇古怪的缩写，头都大了，实在是搞不懂。

其实这些问题也并不是什么新问题，把关键字粘贴进搜索栏，再点一下按钮，搜索引擎马上就能找出几十万个相关的页面。但看完第一页的前几个链接后，通常还是有种“懵懵懂懂”“似懂非懂”的感觉，觉得说的对，又不全对，和自己的思路总是不够“Match”。

不过大多数情况下你可能都没有时间细想，优先目标是把手头的工作“对付过去”。长此以来，你对 HTTP 的认识也可能仅限于这样的“知其然，而不知其所以然”，实际情况就是 HTTP 天天用，时时用，但想认真、系统地学习一下，梳理出自己的知识体系，经常会发现无从下手。

我把这种 HTTP 学习的现状归纳为三点：**正式资料“少”、网上资料“杂”、权威资料“难”**。

第一个，**正式资料“少”**。

上购书网站，搜个 Python、Java，搜个 MySQL、Node.js，能出一大堆。但搜 HTTP，实在是少得可怜，那么几本，一只手的手指头就可以数得过来，和语言类、数据库类、框架类图书真是形成了鲜明的对比。

现有的 HTTP 相关图书我都看过，怎么说呢，它们都有一个特点，“广撒网，捕小鱼”，都是知识点，可未免太“照本宣科”了，理论有余实践不足，看完了还是不知道怎么用。

而且这些书的“岁数”都很大，依据的都是 20 年前的 RFC2616，很多内容都不合时宜，而新标准 7230 已经更新了很多关键的细节。

第二个，**网上资料“杂”**。

正式的图书少，而且过时，那就求助于网络社区吧。现在的博客、论坛、搜索引擎非常发达，网上有很多 HTTP 协议相关的文章，也都是网友的实践经验分享，“干货”很多，很能解决实际问题。

但网上文章的特点是细小、零碎，通常只“钉”在一个很小的知识点上，而且由于帖子长度的限制，无法深入展开论述，很多都是“浅尝辄止”，通常都止步在“How”层次，很少能说到“Why”，能说透的更是寥寥无几。

网文还有一个难以避免的“毛病”，就是“良莠不齐”。同一个主题可能会有好几种不同的说法，有的还会互相矛盾、以讹传讹。这种情况是最麻烦的，你必须花大力气去鉴别真假，不小心就会被“带到沟里”。

可想而知，这种“东一榔头西一棒子”的学习方式，用“碎片”拼凑出来的 HTTP 知识体系是非常不完善的，会有各种漏洞，遇到问题时基本派不上用场，还得再去找其他的“碎片”。

第三个，**权威资料“难”**。

图书少，网文杂，我们还有一个终极的学习资料，那就是 RFC 文档。

RFC 是互联网工程组（IETF）发布的官方文件，是对 HTTP 最权威的定义和解释。但它也是最难懂的，全英文看着费劲，理解起来更是难上加难，文档之间还会互相关联引用，“劝退率”极高。

这三个问题就像是“三座大山”，阻碍了像你这样的很多有心人去学习、了解 HTTP 协议。

那么，如何才能更好地学习 HTTP 呢？

我为这个专栏定了一个基调：**“要有广度，但更要有深度”**。目标是成为含金量最高的 HTTP 学习资料，新手可以由浅入深、系统学习，老手可以温故知新、查缺补漏，让你花最少的时间，用最少的精力，掌握最多、最全面、最系统的知识。

由于 HTTP 应用得非常广泛，几乎涉及到所有的领域，所以我会从广度上从 HTTP 尽量向外扩展，不只讲协议本身，与它相关的 TCP/IP、DNS、SSL/TLS、Web Server 等都会讲到，而且会把它们打通串联在一起，形成知识链，让你知道它们之间是怎么联系、怎么运行的。

专栏文章的深度上我也是下足了功夫，全部基于最新的 RFC 标准文档，再结合我自己多年的实践体会，力求讲清讲透，能让你看了以后有豁然开朗的感觉。

比如分析 HTTPS，我会用 Wireshark 从建立 TCP 连接时就开始抓包，从二进制最底层来分析里面的 Record、Cipher Suite、Extension，讲 ECDHE、AES、SHA384，再画出详细的流程图，做到“一览无余”。

陆游有诗：“纸上得来终觉浅，绝知此事要躬行”。学习网络协议最重要的就是实践，在专栏里我还会教你用 Nginx 搭建一个“麻雀虽小，五脏俱全”的实验环境，让你与 HTTP 零距离接触。

它有一个最大的优点：自身就是一个完整的网络环境，即使不联网也能够在里面收发 HTTP 消息。

我还精心设计了配套的测试用例，最小化应用场景，排除干扰因素，你可以在里面任意测试 HTTP 的各种特性，再配合 Wireshark 抓包，就能够理论结合实践，更好地掌握 HTTP 的知识。

每一讲的末尾，我也会留几个思考题，你可以把它当作是求职时的面试官问题，尽量认真思考后再回答，这样能够把专栏的学习由“被动地听”，转变为“主动地学”，实现“学以致用”。

当然了，你和我的“兴趣点”不可能完全一样，我在讲课时也难免“顾此失彼”“挂一漏万”，希望你积极留言，我会视情况做些调整，或者用答疑的形式补充没讲到的内容。

今年是万维网和 HTTP 诞生 30 周年，也是 HTTP/1.1 诞生 20 周年，套用莎翁《哈姆雷特》里的名句，让我们在接下来的三个月里一起努力。

“To Be a HTTP Hero!”

极客时间

透视 HTTP 协议

深入理解 HTTP 协议本质与应用

罗剑锋

奇虎360技术专家

Nginx/OpenResty 开源项目贡献者



新版升级：点击「请朋友读」，20位好友免费读，邀请订阅更有**现金**奖励。