

## 07 | 自己动手，搭建HTTP实验环境

2019-06-12 Chrono

《透视HTTP协议》

课程介绍 >



讲述：Chrono

时长 09:36 大小 13.19M



这一讲是“破冰篇”的最后一讲，我会先简单地回顾一下之前的内容，然后在 Windows 系统上实际操作，用几个应用软件搭建出一个“最小化”的 HTTP 实验环境，方便后续的“基础篇”“进阶篇”“安全篇”的学习。

### “破冰篇”回顾

HTTP 协议诞生于 30 年前，设计之初的目的是用来传输纯文本数据。但由于形式灵活，搭配 URI、HTML 等技术能够把互联网上的资源都联系起来，构成一个复杂的超文本系统，让人们自由地获取信息，所以得到了迅猛发展。

领资料

HTTP 有多个版本，目前应用的最广泛的是 HTTP/1.1，它几乎可以说是整个互联网的基石。但 HTTP/1.1 的性能难以满足如今的高流量网站，于是又出现了 HTTP/2 和 HTTP/3。不过这两个新版本的协议还没有完全推广开。在可预见的将来，HTTP/1.1 还会继续存在下去。



HTTP 翻译成中文是“超文本传输协议”，是一个应用层的协议，通常基于 TCP/IP，能够在网络的任意两点之间传输文字、图片、音频、视频等数据。

HTTP 协议中的两个端点称为**请求方**和**应答方**。请求方通常就是 Web 浏览器，也叫 user agent，应答方是 Web 服务器，存储着网络上的大部分静态或动态的资源。

在浏览器和服务器之间还有一些“中间人”的角色，如 CDN、网关、代理等，它们也同样遵守 HTTP 协议，可以帮助用户更快速、更安全地获取资源。

HTTP 协议不是一个孤立的协议，需要下层很多其他协议的配合。最基本的是 TCP/IP，实现寻址、路由和可靠的数据传输，还有 DNS 协议实现对互联网上主机的定位查找。

对 HTTP 更准确的称呼是“**HTTP over TCP/IP**”，而另一个“**HTTP over SSL/TLS**”就是增加了安全功能的 HTTPS。

## 软件介绍

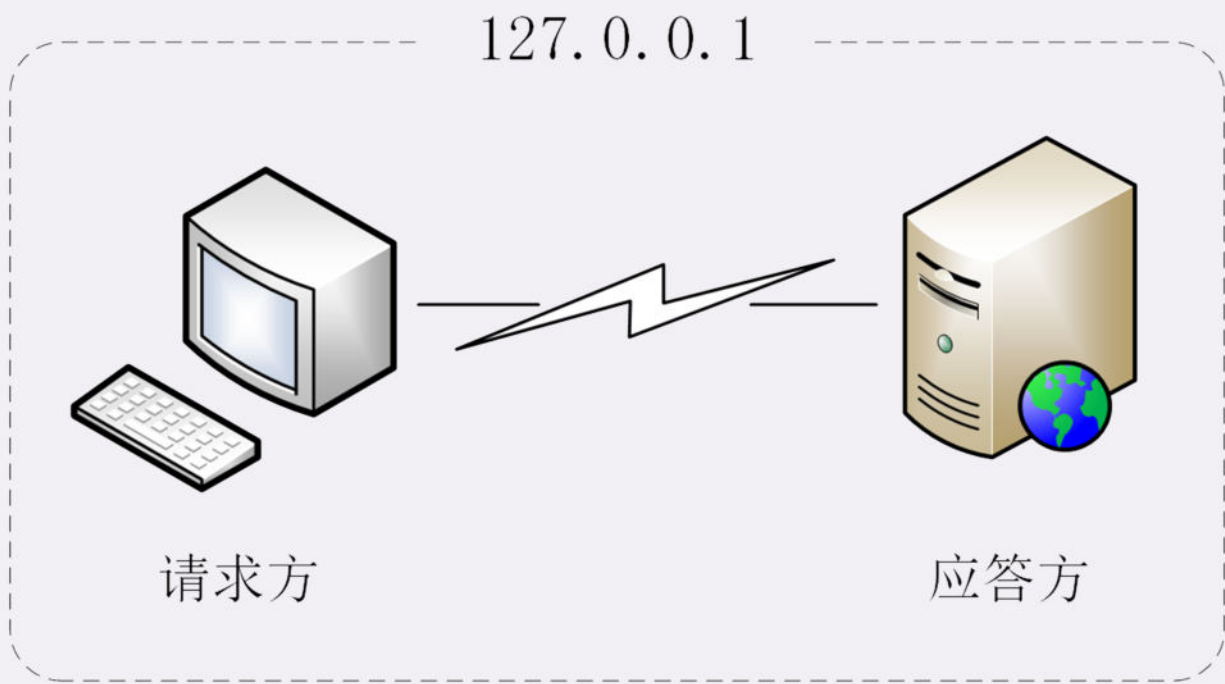
常言道“实践出真知”，又有俗语“光说不练是假把式”。要研究 HTTP 协议，最好有一个实际可操作、可验证的环境，通过实际的数据、现象来学习，肯定要比单纯的“动嘴皮子”效果要好的多。

现成的环境当然有，只要能用浏览器上网，就会有 HTTP 协议，就可以进行实验。但现实的网络环境又太复杂了，有很多无关的干扰因素，这些“噪音”会“淹没”真正有用的信息。

所以，我给你的建议是：搭建一个“**最小化**”的环境，在这个环境里仅有 HTTP 协议的两个端点：请求方和应答方，去除一切多余的环节，从而可以抓住重点，快速掌握 HTTP 的本质。

领资料





简单说一下这个“最小化”环境用到的应用软件：

- Wireshark
- Chrome/Firefox
- Telnet
- OpenResty

**Wireshark** 是著名的网络抓包工具，能够截获在 TCP/IP 协议栈中传输的所有流量，并按协议类型、地址、端口等任意过滤，功能非常强大，是学习网络协议的必备工具。

它就像是网络世界里的一台“高速摄像机”，把只在一瞬间发生的网络传输过程如实地“拍摄”下来，事后再“慢速回放”，让我们能够静下心来仔细地分析那一瞬到底发生了什么。

**Chrome** 是 Google 开发的浏览器，是目前的主流浏览器之一。它不仅上网方便，也是一个很好的调试器，对 HTTP/1.1、HTTPS、HTTP/2、QUIC 等的协议都支持得非常好，用 F12 打开“开发者工具”还可以非常详细地观测 HTTP 传输全过程的各种数据。

如果你更习惯使用 **Firefox**，那也没问题，其实它和 Chrome 功能上都差不太多，选择自己喜欢的就好。

领资料



与 Wireshark 不同，Chrome 和 Firefox 属于“事后诸葛亮”，不能观测 HTTP 传输的过程，只能看到结果。

**Telnet** 是一个经典的虚拟终端，基于 TCP 协议远程登录主机，我们可以使用它来模拟浏览器的行为，连接服务器后手动发送 HTTP 请求，把浏览器的干扰也彻底排除，能够从最原始的层面去研究 HTTP 协议。

**OpenResty** 你可能比较陌生，它是基于 Nginx 的一个“强化包”，里面除了 Nginx 还有一大堆有用的功能模块，不仅支持 HTTP/HTTPS，还特别集成了脚本语言 Lua 简化 Nginx 二次开发，方便快速地搭建动态网关，更能够当成应用容器来编写业务逻辑。

选择 OpenResty 而不直接用 Nginx 的原因是它相当于 Nginx 的“超集”，功能更丰富，安装部署更方便。我也会用 Lua 编写一些服务端脚本，实现简单的 Web 服务器响应逻辑，方便实验。

## 安装过程

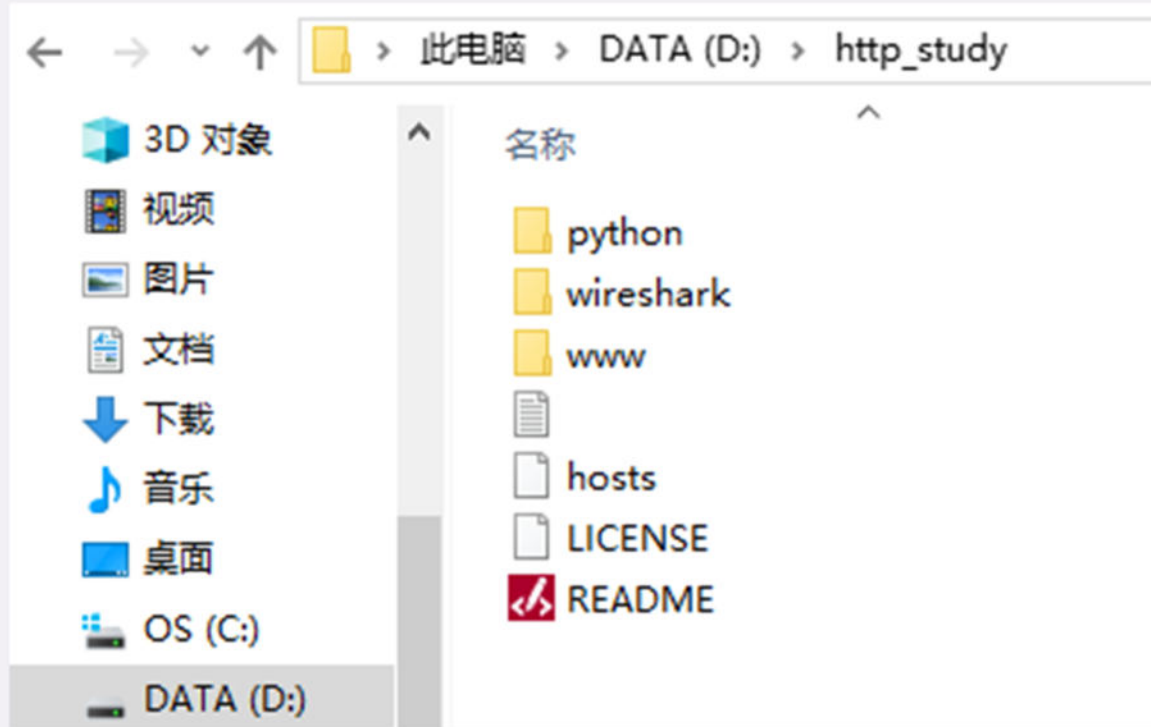
这个“最小化”环境的安装过程也比较简单，大约只需要你半个小时不到的时间就能搭建完成。

我在 GitHub 上为本专栏开了一个项目：[🔗http\\_study](#)，可以直接用“git clone”下载，或者去 Release 页面，下载打好的[🔗压缩包](#)。

我使用的操作环境是 Windows 10，如果你用的是 Mac 或者 Linux，可以用 VirtualBox 等虚拟机软件安装一个 Windows 虚拟机，再在里面操作（或者可以到“答疑篇”的[🔗Linux/Mac 实验环境搭建](#)中查看搭建方法）。

首先你要获取**最新**的 http\_study 项目源码，假设 clone 或解压的目录是“D:\http\_study”，操作完成后大概是下图这个样子。

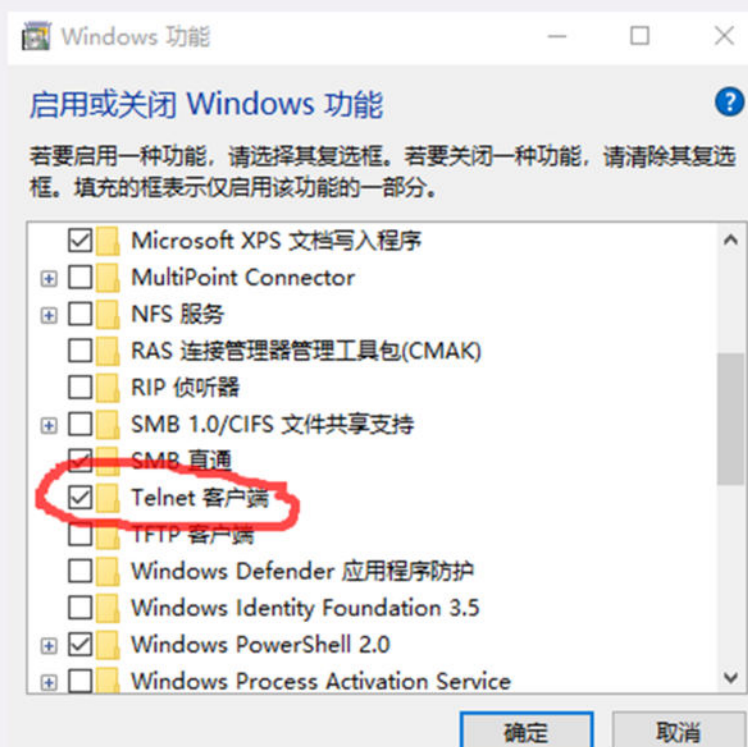




Chrome 和 WireShark 的安装比较简单，一路按“下一步”就可以了。版本方面使用最新的就好，我的版本可能不是最新的，Chrome 是 73，WireShark 是 3.0.0。

Windows 10 自带 Telnet，不需要安装，但默认是不启用的，需要你稍微设置一下。

打开 Windows 的设置窗口，搜索“Telnet”，就会找到“启用或关闭 Windows 功能”，在这个窗口里找到“Telnet 客户端”，打上对钩就可以了，可以参考截图。



领资料

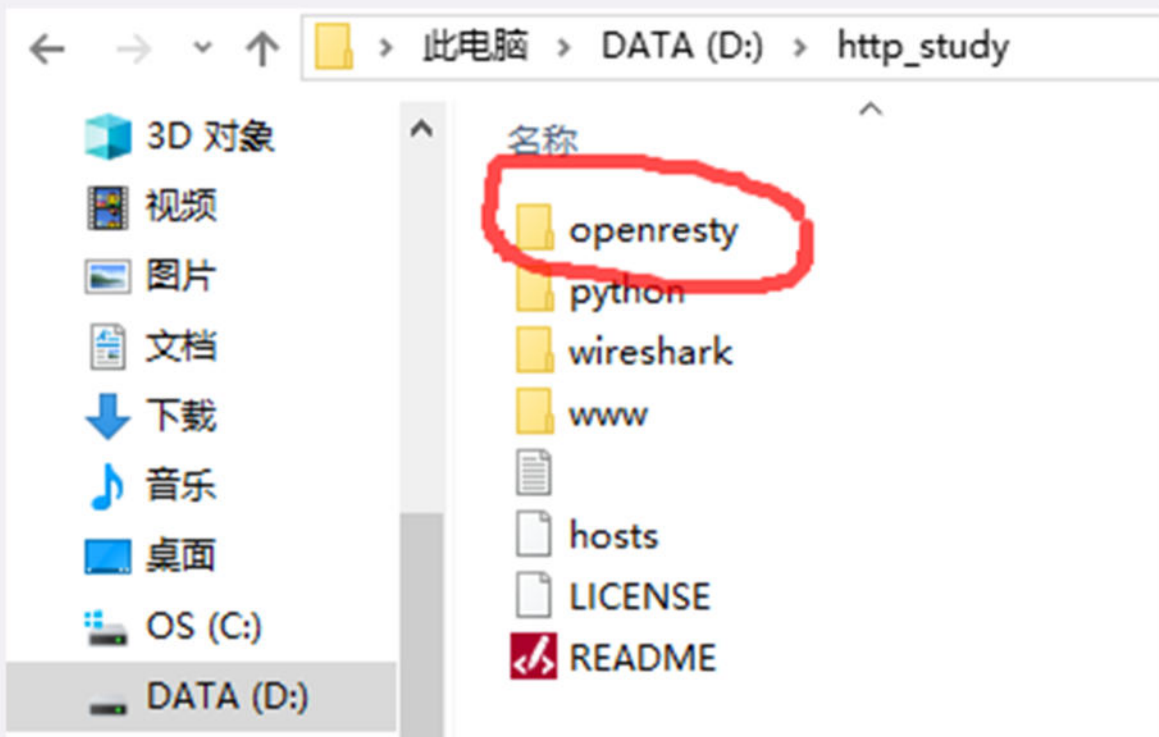




接下来我们要安装 OpenResty，去它的 [官网](#)，点击左边栏的“Download”，进入下载页面，下载适合你系统的版本（这里我下载的是 64 位的 1.15.8.1，包的名字是“openresty-1.15.8.1-win64.zip”）。

- For 32-bit Windows: [openresty-1.15.8.1-win32.zip](#)  
12MB [PGP](#) - 14 May 2018
- For 64-bit Windows: [openresty-1.15.8.1-win64.zip](#)  
12MB [PGP](#) - 14 May 2018

然后要注意，你必须把 OpenResty 的压缩包解压到刚才的“D:\http\_study”目录里，并改名为“openresty”。



安装工作马上就要完成了，为了能够让浏览器能够使用 DNS 域名访问我们的实验环境，还要改一下本机的 hosts 文件，位置在“C:\WINDOWS\system32\drivers\etc”，在里面添加三行本机 IP 地址到测试域名的映射，你也可以参考 GitHub 项目里的 hosts 文件，这就相当于一台物理实机上“托管”了三个虚拟主机。



领资料

```
1 127.0.0.1      www.chrono.com
2 127.0.0.1      www.metroid.net
3 127.0.0.1      origin.io
```

注意修改 hosts 文件需要管理员权限，直接用记事本编辑是不行的，可以切换管理员身份，或者改用其他高级编辑器，比如 Notepad++，而且改之前最好做个备份。

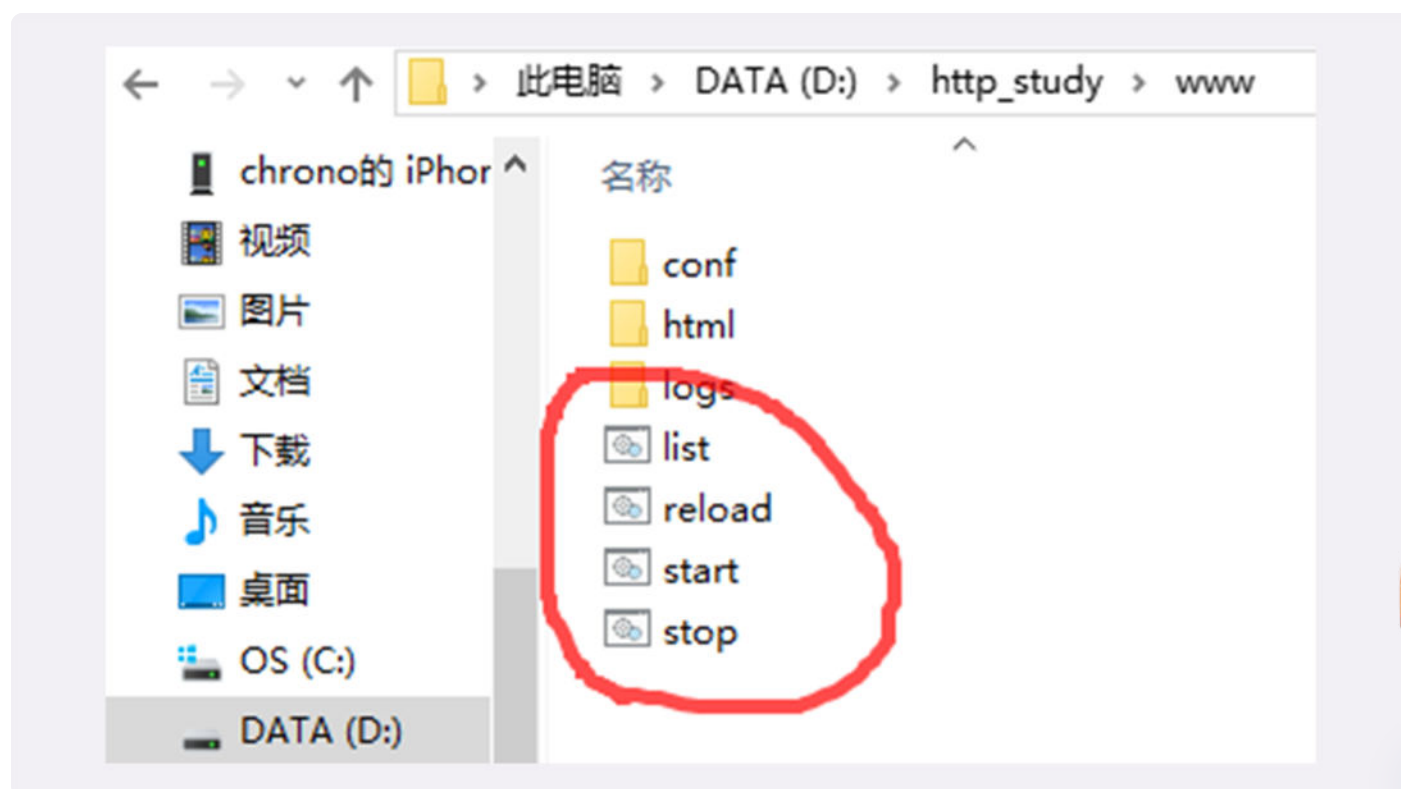
到这里，我们的安装工作就完成了！之后你就可以用 Wireshark、Chrome、Telnet 在这个环境里随意“折腾”，弄坏了也不要紧，只要把目录删除，再来一遍操作就能复原。

## 测试验证

实验环境搭建完了，但还需要把它运行起来，做一个简单的测试验证，看是否运转正常。

首先我们要启动 Web 服务器，也就是 OpenResty。

在 http\_study 的“www”目录下有四个批处理文件，分别是：



领资料



- start：启动 OpenResty 服务器；
- stop：停止 OpenResty 服务器；

- reload: 重启 OpenResty 服务器;
- list: 列出已经启动的 OpenResty 服务器进程。

使用鼠标双击“start”批处理文件，就会启动 OpenResty 服务器在后台运行，这个过程可能会有 Windows 防火墙的警告，选择“允许”即可。

运行后，鼠标双击“list”可以查看 OpenResty 是否已经正常启动，应该会有两个 nginx.exe 的后台进程，大概是下图的样子。

```
D:\http_study\www>tasklist /fi "imagename eq nginx.exe"
```

映像名称	PID	会话名	会话#	内存使用
nginx.exe	35088	Console	1	8,928 K
nginx.exe	39988	Console	1	9,524 K

```
D:\http_study\www>pause  
请按任意键继续. . .
```

有了 Web 服务器后，接下来我们要运行 Wireshark，开始抓包。

因为我们的实验环境运行在本机的 127.0.0.1 上，也就是 loopback“环回”地址。所以，在 Wireshark 里要选择“Npcap loopback Adapter”，过滤器选择“HTTP TCP port(80)”，即只抓取 HTTP 相关的数据包。鼠标双击开始界面里的“Npcap loopback Adapter”即可开始抓取本机上的网络数据。

欢迎使用 Wireshark

捕获

...使用这个过滤器:

显示所有接口

Npcap Loopback Adapter

本地连接\* 11

本地连接\* 10

本地连接\* 9

以太网

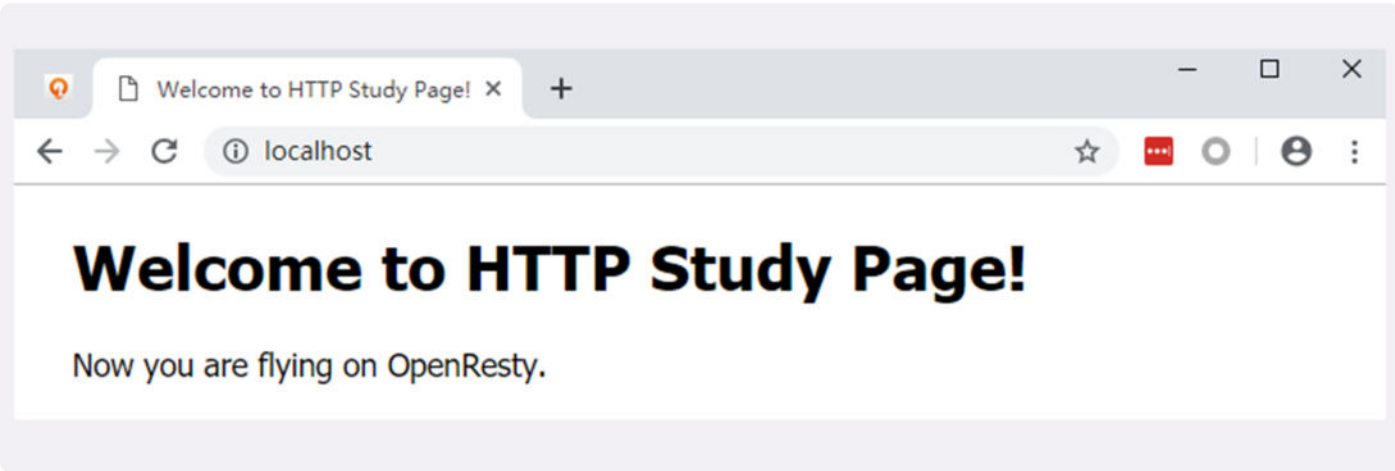
蓝牙网络连接

领资料

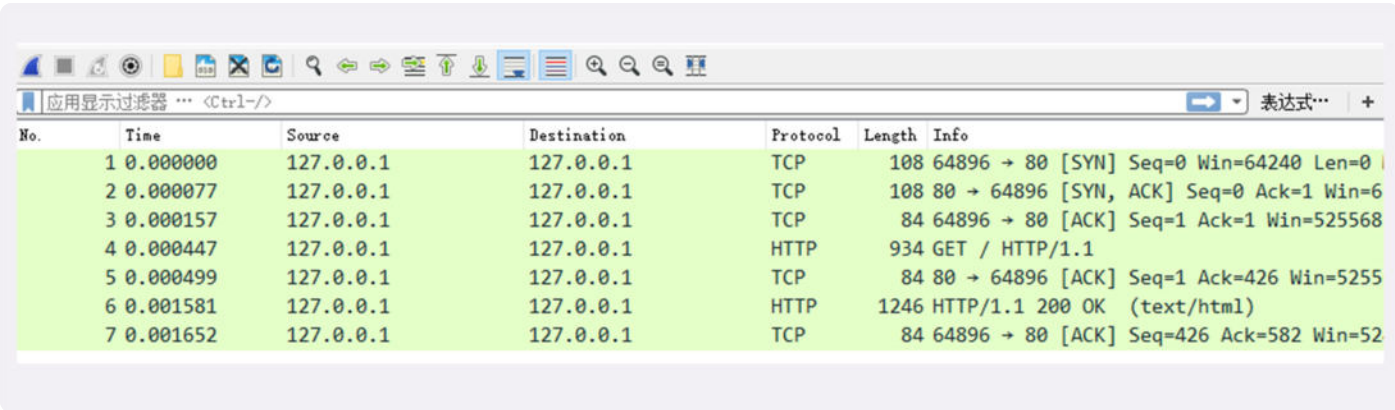




然后我们打开 Chrome，在地址栏输入“http://localhost”，访问刚才启动的 OpenResty 服务器，就会看到一个简单的欢迎界面，如下图所示。



这时再回头去看 Wireshark，应该会显示已经抓到了一些数据，就可以用鼠标点击工具栏里的“停止捕获”按钮告诉 Wireshark“到此为止”，不再继续抓包。



至于这些数据是什么，表示什么含义，我会在下一讲再详细介绍。

如果你能够在自己的电脑上走到这一步，就说明“最小化”的实验环境已经搭建成功了，不要忘了实验结束后运行批处理“stop”停止 OpenResty 服务器。

### 小结

这次我们学习了如何在自己的电脑上搭建 HTTP 实验环境，在这里简单小结一下今天的内容。

1. 现实的网络环境太复杂，有很多干扰因素，搭建“最小化”的环境可以快速抓住重点，掌握 HTTP 的本质；

领资料



2. 我们选择 Wireshark 作为抓包工具，捕获在 TCP/IP 协议栈中传输的所有流量；
3. 我们选择 Chrome 或 Firefox 浏览器作为 HTTP 协议中的 user agent；
4. 我们选择 OpenResty 作为 Web 服务器，它是一个 Nginx 的“强化包”，功能非常丰富；
5. Telnet 是一个命令行工具，可用来登录主机模拟浏览器操作；
6. 在 GitHub 上可以下载到本专栏的专用项目源码，只要把 OpenResty 解压到里面即可完成实验环境的搭建。

## 课下作业

1. 按照今天所学的，在你自己的电脑上搭建出这个 HTTP 实验环境并测试验证。
2. 由于篇幅所限，我无法详细介绍 Wireshark，你有时间可以再上网搜索 Wireshark 相关的资料，了解更多的用法。

欢迎你把自己的学习体会写在留言区，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。



## 课外小贴士

- 01 如果你会编程，还可以选择一种自己擅长的语言（比如 Python），调用专用库，去访问 OpenResty 服务器。
- 02 在 Linux 上可以直接从源码编译 OpenResty，用 curl 发送测试命令，用 tcpdump 抓包。

领资料



- 03 除了经典的 Wireshark，另外有一个专门抓 HTTP 包的工具 Fiddler。
- 04 如果无法正常启动 OpenResty，最大的可能就是端口 80 或 443 被占用了（比如安装了 VMWare workstation），先看“www/logs”里的错误日志，然后在命令行里用“netstat -aon | findstr :443”找到占用的进程或服务，手动停止就可以了。
- 05 有的时候可能 stop 批处理无法正确停止 OpenResty，你可以用“任务管理器”在后台进程里查找 nginx.exe，然后手动强制关闭。

# 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。

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 21     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   06 | 域名里有哪些门道？

下一篇   08 | 键入网址再按下回车，后面究竟发生了什么？

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 (110)

写留言



cylim

2019-06-12

在Mac上,

拷贝项目 (需要Git)

1. git clone [https://github.com/chronolaw/http\\_study](https://github.com/chronolaw/http_study)

安装OpenResty (推荐使用Homebrew)

1. brew tap openresty/brew

2. brew install openresty

运行项目

1. cd http\_study/www/

2. openresty -p `pwd` -c conf/nginx.conf

停止项目

1. openresty -s quit -p `pwd` -c conf/nginx.conf

作者回复: 好同学!! 赞!

领资料







古夜

2019-06-12

我打赌很多人抓不到包，找不到本地回环地址，不知道最新版的wireshark是否修复了这个问题，如果出现以上问题，记得卸载重装wireshark，不要勾选它自带的ncap应该是这个名字，然后自己去单独下一个这个软件

作者回复: 有问题欢迎提出来，我机器上的Wireshark装的比较早，具体的步骤记不太清了，应该是很简单的。

共 7 条评论 >

👍 21



Fstar

2019-06-28

对 cylim 的 mac 上运行 openresty 的教程进行补充：

按照 cylim 的做法，我遇到了访问 localhost 时，网页报 403 错误的情况，原因是没有 html/index.html 文件的访问权限。我研究并找到了解决方案：

先 `ls -la html`，查看文件的权限，得到 user 和 group，我这里是 fstar 和 staff。

然后在 `conf/nginx.conf` 文件的顶部添加

`user fstar staff;`

然后再启动 openresty 就可以正常访问了。

作者回复: 感谢同学的热心补充。

共 4 条评论 >

👍 16



名曰蓝兮

2019-06-19

centos上的安装步骤，有错误请指出  
wireshark：

1. `yum install wireshark`

`yum install wireshark-gnome`

2. 如果不是root用户，启动后没有权限，做如下操作

2.1 添加当前用户到wireshark组，我的用户叫'zp'：

`usermod -a -G wireshark zp`

2.2 然后给dumpcap读网卡的权限：

`setcap cap_net_raw,cap_net_admin+eip /usr/sbin/dumpcap`

完成后重启机器。

领资料



telnet:

yum install telnet

OpenResty:

官网有说明，按照说明一步步来

1. 添加OpenResty仓库:

```
sudo yum install yum-utils
```

```
sudo yum-config-manager --add-repo https://openresty.org/package/centos/openresty.repo
```

2. 安装OpenResty:

```
sudo yum install openresty
```

```
sudo yum install openresty-resty
```

3. 在~目录下创建conf和logs文件夹:

```
mkdir ~/work
```

```
cd ~/work
```

```
mkdir logs/ conf/
```

4. 在conf文件夹下创建nginx.conf文件，内容如下:

```
worker_processes 1;
```

```
error_log logs/error.log;
```

```
events {
```

```
    worker_connections 1024;
```

```
}
```

```
http {
```

```
    server {
```

```
        listen 8080;
```

```
        location / {
```

```
            default_type text/html;
```

```
            content_by_lua_block {
```

```
                ngx.say("<p>hello, world</p>")
```

```
            }
```

```
        }
```

```
    }
```

```
}
```

5. 添加OpenResty环境变量，注意冒号，别丢了:

```
PATH=/usr/local/openresty/nginx/sbin:$PATH
```

```
export PATH
```

6. 在'~/work'目录下启动OpenResty:

```
nginx -p `pwd`/ -c conf/nginx.conf
```

7. 验证安装:

```
curl http://localhost:8080
```

领资料



输出：  
<p>hello, world</p>

作者回复: 写的很详细, 赞!



11



pyhhou

2019-06-12

想请问下在 MacOS 或者是 Linux 上怎么搭建? (不是太想弄 Windows 虚拟机)

作者回复: 需要用brew或者yum安装OpenResty, 然后看一下nginx.conf, 里面的注释有说明。

共 2 条评论 >

11



郁方林

2019-06-12

start启动完成后, cmd窗口一闪而过, 当我点击list启动时显示“没有运行的任务匹配制定标准”, 请按任意键继续, 当我随便输入数据时, cmd窗口又没了

作者回复: 看一下www/logs/error.log, 是否有端口被占用了。

共 2 条评论 >

9



珈蓝白塔

2020-04-13

Mac 开发环境的搭建参考《答疑篇41》, 项目中已经有前辈写好的 shell 脚本, 终端里直接运行就可以, 不需要自己输入 openresty 命令啦; 服务器启动以后访问 localhost 环境遇到了 403 问题, 显示不出来 HTML, 可参照留言区中提出的, 在 conf/nginx.conf 文件的顶部添加 user xxxx staff; 来解决, 这个 xxxx 是自己的 mac 账户名; Wireshark(v3.2.3) 中选择环回地址时, 选择 lo: lo 就可以啦, 过滤器是和文中一样的, 已成功搭建环境 (2020 年4月13日)

作者回复: 欢迎经验分享, 让同学都少走弯路。

共 3 条评论 >

8



Leon

2019-06-12

破冰篇最后一篇, 是马上开展破冰行动, 抓捕林耀东了吗

领资料



作者回复: 写这个的时候电视剧还没出呢，完全的碰巧，笑。



👍 8



**YUANWOW**

2019-07-02

我一开始nginx一直起不来  
后面看了error.log  
发现本机443端口被占用了  
netstat -ano | findstr "443"  
看到一个 0.0.0.0:443 最后一列是进程的PID  
查找到是vmware-hostd这个进程  
后面谷歌搜索了下  
vmware的虚拟机共享会默认占用443端口  
所以安装了vmware的把虚拟机共享关闭就好了

作者回复: 欢迎经验分享。

共 3 条评论 >

👍 7



**Amark**

2019-06-12

老师，上面过程怎么没有用到telnet

作者回复: 后面会用，Telnet需要手动输入http请求，比较麻烦，只有在比较特殊的时候才会用。



👍 6



**Geek\_d4dee7**

2019-06-12

老师 最近我维护的一个网站打开速度非常慢 服务器CPU 负载0.5到0.8之间 有十多台web 服务器 redis db 负载都正常 只是nginx 的连接数在出问题的时间点有上升 我目前不知道从哪下手排查这个问题 是用php symfony 开发的 能否给点思路 万分感谢

作者回复: 在日志里加上\$upstream\_connect\_time、\$upstream\_header\_time、\$upstream\_response\_time这几个变量，看看反向代理耗时在哪里。

另外也可以用systemtap，抓火焰图看看。



👍 6

领资料





geek桃

2021-03-03

送给后来的同学：

如果你按照步骤操作之后出现：start启动完成后，cmd窗口一闪而过，点击list启动时显示“没有运行的任务匹配制定标准”，请按任意键继续，当随便输入数据时，cmd窗口又没了；去查找www/logs/error.log，如果日志报错为“10013: An attempt was made to access a socket in a way forbidden by its access permissions”，说明你的80端口被占用了，按照下面步骤操作。

- 1.按键盘win+r 打开运行界面，输入cmd，确定，打开管理员界面
- 2.输入 netstat -aon | findstr :80 （有一条0.0.0.0的数据，记住这条数据最后的数字；我的是5884）
- 3.输入 tasklist|findstr "5884" （根据上一步查到的数字，找到5884端口对应的服务名称，我的是snv）
- 4.在控制台关闭服务
- 5.重新启动start.bat，成功！

作者回复: 非常好的经验分享，鼓励。

Windows环境比较复杂，容易出各种错误，如果不好解决可以尝试用虚拟机或者docker。

共 3 条评论 >

👍 5



Cris

2019-07-11

在浏览器和服务器之间还存在“中间人”，这些中间人也都遵循http协议，我想问下，这些中间人是不是都工作在应用层？

作者回复: 是的，都是用http协议，当然就是在应用层。



👍 4



QQ怪

2019-06-12

为啥有时候批处理stop不掉openresty?

作者回复: 可能是多次start，stop就失效了，只能手动在任务管理器里关闭。



👍 4



Leon

2019-06-12

领资料





老师可以把环境打包成容器，我们进容器直接嗨，隔离更彻底

作者回复: 考虑大多数同学都用的是Windows，所以暂时只能这样，手动操作也能加深一下印象吧。



👍 4



兔嘟嘟

2021-07-16

windows上装的最新的3.4.7版，没有npcap，但是有一个Adapter for loopback traffic，用起来效果一样，就是抓到的Source是::1，猜测是这台电脑比较新，用上了IPv6的localhost

作者回复: 这篇文章已经是两年前的了，随着软件的升级，可能有的选项已经过时了，欢迎同学们随时更新。

共 2 条评论 >

👍 3



不是云不飘

2019-06-17

建议还是能有win和Mac，逼近做开发的Mac不再少数。这些东西之前只有客户对接问题才会看到运维大哥在哪捣腾那时候看的一脸们逼，难得如此细致的了解。

作者回复: 有同学已经写的很详细了，看看后续是否再专门详细写一下Linux和mac的搭建吧。



👍 3



6欢

2019-06-12

建议环境搭建都在linux操作，哈哈

作者回复: 我也是这么想，可惜用Windows的同学还是不少。



👍 3



sunözil

2019-07-05

希望有个Mac环境搭建 谢谢老师

作者回复: 已经有同学回复了，比较详细，有不清楚的可以再问。



👍 2

领资料





bywuu

2019-06-13

成功了！这里需要下载wireshark，不过下载之后最好更新为最新版本3.0.x（最好翻墙），否则最好是重启，否则看不到。如果是先打开了localhost，那么应该刷新一下，才能在wireshark里面看到结果。

运行了stop脚本之后，再刷新浏览器，就会提示找不到页面了。这时的wireshark里面也都是红黑色的出错信息了。

作者回复: 辛苦终有回报。



领资料

