数据包的寿命

互联网的4层模型从应用层获取数据流。传输层将这一数据流分成若干数据段，可靠地传递给另一台计算机上运行的应用程序。传输层将这些数据段作为网络层数据包发送，由网络层传递给另一台计算机。让我们看看在实践中是什么样子的，一个网络浏览器和服务器发送和接收的实际数据包。

TCP字节流

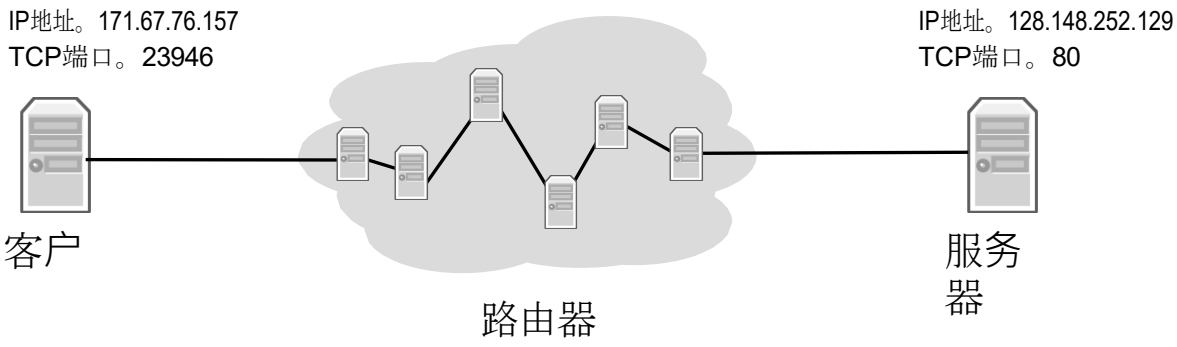
首先，让我们看一下传输层。几乎所有的网络流量都是通过TCP，即传输控制协议。在其典型操作中，有一个客户和一个服务器。服务器监听连接请求。为了打开一个连接，客户端发出一个连接请求，服务器对其进行响应。我不会去研究这个工作的具体细节，但事实证明，这种交换需要三条信息，称为 "三方握手"。

握手的第一步是客户端向服务器发送一个 "同步 "消息，通常称为SYN。第二步是服务器响应一个 "同步 "消息，同时确认客户的 "同步"，或一个 "同步和确认消息"，通常称为SYN-ACK。第三步，也是最后一步，是客户通过确认服务器的同步来作出回应，通常称为ACK。因此，三向握手通常被描述为 "同步、同步和确认、确认"，或 "SYN、SYN-ACK、ACK"。

TCP字节流

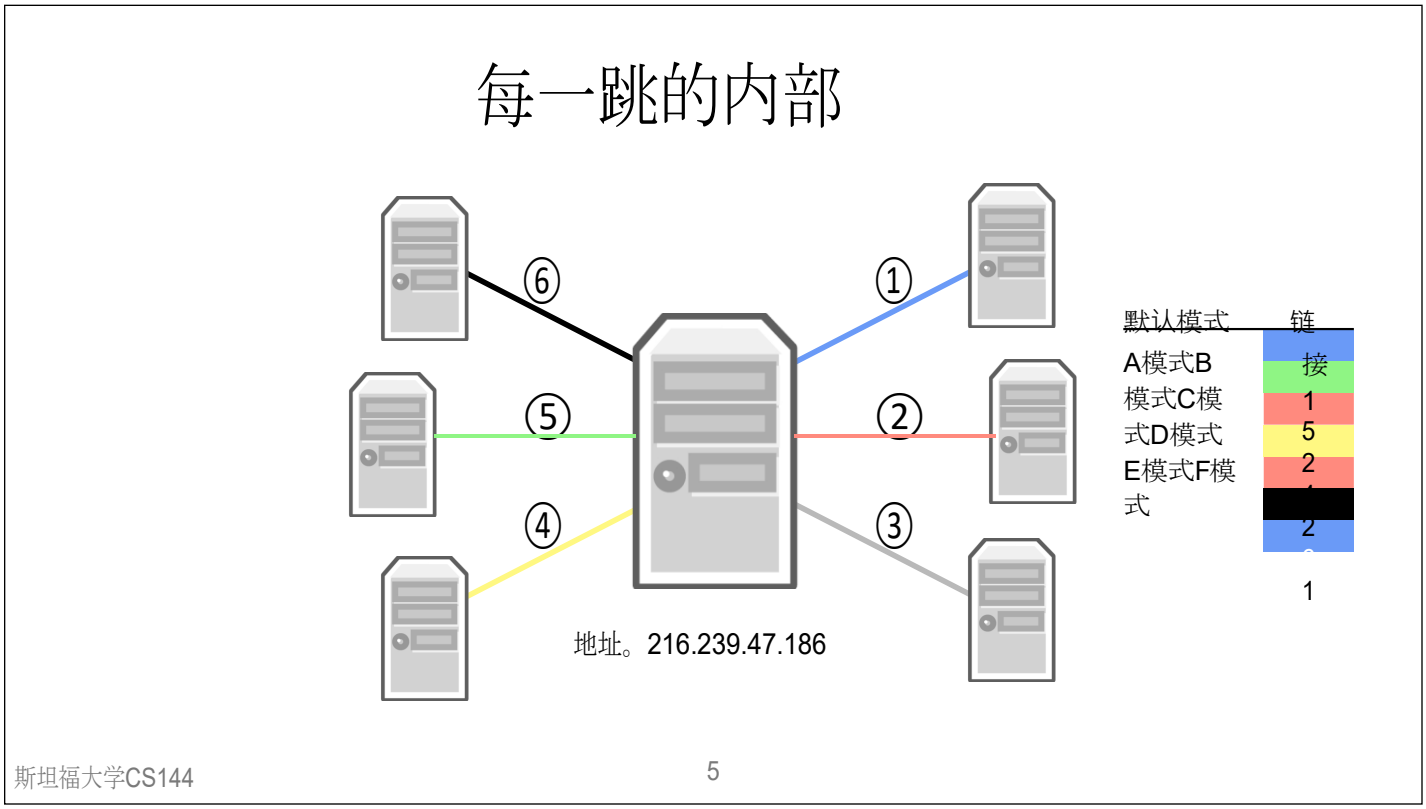
回顾一下，网络层负责向计算机传递数据包，但传输层负责向应用程序传递数据。从网络层的角度来看，发送到同一台计算机上不同应用程序的数据包看起来是一样的。这意味着，要打开一个TCP流到另一个程序，我们需要两个地址。第一个是互联网协议地址，是网络层用来向计算机传送数据包的地址。第二个是TCP端口，告诉计算机的软件要把数据传送给哪个应用程序。网络服务器通常在TCP 80端口运行。因此，当我们打开一个与网络服务器的连接时，我们向运行网络服务器的计算机发送IP数据包，其目标地址是该计算机的IP地址。这些IP数据包有TCP段，其目的端口是80。

溪流内部



但这些IP数据包是如何到达目的地的呢？我们没有一条直接连接我的客户和服务器的线路。相反，我的客户机被连接到一个中间计算机，即一个路由器。这个路由器本身又与其他路由器相连。客户端和服务器之间的IP数据包需要经过许多 "跳"，其中一个跳是连接两个路由器的链接。例如，由于我的客户是在一个WiFi网络上，第一跳是无线到WiFi接入点。接入点有一个与更广泛的互联网的有线连接，所以它沿这个有线跳转我的客户的数据包。

一个路由器可以有许多连接它的链接。当每个数据包到达时，路由器决定用哪条链路将其发送出去。路由器有IP地址，所以它也有可能不转发数据包，而是将其送到自己的软件中。例如，当你使用TCP登录路由器时，IP数据包的目的地是路由器自己的IP地址。



路由器是如何做出这一决定的？它通过一个叫做转发表的东西来做，在这里显示在右边。转发表由一组IP地址模式和为每个模式发送的链接组成。

当一个数据包到达时，路由器检查哪个转发表项的模式最符合该数据包。它沿着该条目的链接转发该数据包。一般来说，"最佳 "意味着最具体的匹配。我将在关于最长前缀匹配的视频中更详细地描述这种匹配的原理。但在这个简单的例子中，我们只考虑默认路由，即上表中的第一个条目。默认路由是最不具体的路由--它匹配每一个IP地址。如果当一个数据包到达时，没有比默认路由更具体的路由，路由器将只使用默认路由。

默认路由在边缘网络中特别有用。比如说，你是斯坦福大学，有一个路由器把你和更大的互联网连接起来。你的路由器将为斯坦福大学的网络的IP地址设置许多具体的路由。"通过这个链接将数据包发送到工程学院"，"通过那个链接将数据包发送到图书馆"。但是，如果目标IP地址不是斯坦福大学的，那么路由器应该把它发送到更大的互联网。

引擎盖下

- 从www.cs.brown.edu 要求网页
- 使用wireshark查看TCP字节流的建立和数据交换
- 使用traceroute查看数据包通过互联网的路线

所以现在让我们看看真实网络中的一些IP数据包。我将从www.brown.edu，并使用一个叫做Wireshark的工具向你展示所有的数据包。我们将看到我的网络浏览器是如何通过SYN、SYN-ACK、ACK的三次握手来打开与布朗网络服务器的TCP连接，然后开始发出HTTP GET请求，服务器用数据进行回应。一旦我们看到我的客户端和布朗大学网络服务器之间的数据包交换，我将使用另一个工具，称为traceroute，来观察这些数据包在互联网上的路径。

引擎盖下

- 从www.cs.brown.edu 要求网页
- 使用wireshark查看TCP字节流的建立和数据交换
- 使用traceroute查看数据包通过互联网的路线

因此，首先我将启动wireshark。因为我的电脑正在使用许多网络应用程序，并发送许多不同的数据包，我将告诉wireshark只显示使用80端口的布朗服务器的TCP段数据包。这样，我们就只能看到我产生的网络流量。我还将告诉Wireshark监听en1，这是我的Mac给我的WiFi链路层的名字。正如你所看到的，我有许多可用的链接层，但让我们只看en1，因为这是我与互联网连接的方式。

接下来，我将打开我的网络浏览器，请求浏览布朗大学计算机科学系的网页。这是我本科就读的地方，所以我想了解一下该系的新闻。你可以在wireshark中看到，加载这个网页涉及到发送和接收大量的数据包!
Wireshark向我显示了每个数据包的时间戳、源IP地址、目的IP地址、它使用的协议、它的长度以及进一步的信息。看看这第一个数据包。它是从我的计算机（地址为192.168.0.106）到布朗CS的网络服务器（地址为128.148.32.12）。它将进入TCP 80端口--服务器上的超文本传输协议端口，我们可以从信息栏中的> http字段看到。这些数据包是SYN数据包，是三方握手的第一步。

请看前三个数据包。第一个是一个从我的电脑到网络服务器的SYN数据包。第二个是一个SYN-ACK数据包，从网络服务器返回到我的电脑。第三个是一个从我的计算机返回到网络服务器的ACK。这就是三方握手的过程。现在两台电脑可以交换数据了，你可以看到第一个数据包是一个HTTP请求--我的电脑向网络服务器发送了一个GET请求。对这个GET请求的响应是三个数据包 -- wireshark显示了它收到第三个数据包时的响应，显示在信息为HTTP/1.1 200 OK的那一行。因此，在这里我们可以看到我是如何从布朗的计算机科学服务器上请求一个网页的，通过3个IP数据包的3-way handshake创建一个TCP连接，然后是更多的HTTP请求和响应数据包。

引擎盖下

- 从www.cs.brown.edu 要求网页
- 使用wireshark查看TCP字节流的建立和数据交换
- 使用traceroute查看数据包通过互联网的路线

这就是网络在终端主机（即计算机）看来的样子，因为它们在网络层交换数据包。但是，在网络层内部是什么样子的呢？这些数据包要经过哪些跳数？为了看到这一点，我将使用第二个工具，Traceroute。Traceroute显示数据包到一个目的地的跳数。所以我们可以输入traceroute www.cs.brown.edu 来查看通过互联网的路径。我将添加-w标志，它指定一个超时，超时时间为1秒。

数据包的第一跳是到我的无线路由器，其IP地址是192.168.0.1。从下一跳可以看出，我在家里 -- 我有一个电缆调制解调器，我的互联网供应商是Astound。在这之后，数据包再跳到一个IP地址为74.14.0.3的路由器。之后的一跳是加州旧金山的一个路由器，然后是圣何塞的几个路由器，sjc代表overly.net，sanjose1代表level3.net。在sanjose1.level3.net之后，数据包穿过美国的纽约!它们在纽约经过一系列的路由器--ebr、csw、ebr，然后，在第13跳，到达波士顿。波士顿离布朗大学所在的普罗维登斯非常近。在oshean.org之后，我们看到三颗星--这意味着有一个路由器不会告诉traceroute自己的情况。这些星星是traceroute的方式，显示它在等待回复，但回复超时了。在第20跳，我们看到Brown的CS部门有一个路由器。在那之后，所有的东西都被隐藏起来了--布朗的CS部门不希望你能看到它的网络内部是什么样子的。

因为我们无法看到布朗的网络服务器的路径终点，让我们尝试另一个：麻省理工学院的计算机科学和人工智能实验室（CSAIL）。我们可以看到，数据包到波士顿的路径是一样的，直到第15跳。通往布朗的路径在第15跳进入oshean，而通往麻省理工学院的路径在level3的网络中继续。在通往www.csail.mit.edu 的路径上，只有两个路由器被隐藏，即第13跳和第19跳。我们可以看到，www.csail.mit.edu 也被命名为akron.csail.mit.edu，并且在22跳之后，从我的电脑发出的数据包到达了MIT的网络服务器。看看时间值 -- 我

的数据包到达麻省理工学院网络服务器和它的响应返回给我的时间 -- 来回的时间 -- 不到90毫秒，或不到一个眼睛眨动的时间。

引擎盖下

- 从www.cs.brown.edu 要求网页
- 使用wireshark查看TCP字节流的建立和数据交换
- 使用traceroute查看数据包通过互联网的路线

我们现在已经看到了一个数据包的生命，从一个应用级的客户网络请求开始，通过互联网进行了近20次跳跃，到达目的地。对我来说，这是教授这门课程最好的事情之一。我们所介绍的一切，都是你和我每天都在互动的东西--甚至只是在看一个视频的空间里！我们很容易就能看到原理。在实践中很容易看到这些原则和想法，而且通过一些简单的工具，你可以实时检查互联网的运行情况。