

目录

1. 介绍
2. 从哪里开始？互联网地址
3. 协议堆栈和数据包
4. 网络基础设施
5. 互联网基础设施
6. 互联网路由层次结构
7. 域名和地址解析
8. 重新审视互联网协议
9. 应用协议：HTTP和万维网
10. 应用协议：SMTP和电子邮件
11. 传输控制协议
12. 互联网协议
13. 总结
14. 资源
15. 参考书目

1. 介绍

互联网是怎么工作的？这是个好问题！互联网的发展已成为爆炸性的增长，似乎无法摆脱 `www.com` 的轰炸，在新闻中、手机中看到。由于互联网已经成为我们生活不可或缺的一部分，因此需要充分的了解，次啊能最有效的使用这一工具。

本文章解释了使互联网正常工作的底层基础设施和技术。它没有深入，但涵盖了足够的每个领域，以对所涉及的概念有一个基本的了解。对于没有涉及的问题，文章末尾提供了资源列表。

2. 从哪里开始？互联网地址

因为互联网是一个全球计算机网络，连接到互联网的每台计算机都必须有一个唯一的地址。互联网地址的格式是 `nnn.nnn.nnn.nnn`，其中nnn必须是0到255之间的数字。此地址成为 **IP地址**。（IP代表Internet协议；下面会详细介绍）

下图说明了两台连接到互联网的计算机；IP地址为 1.2.3.4 的计算机和另一台IP地址 5.6.7.8的计算机。互联网被表示为介于两者之间的抽象对象。

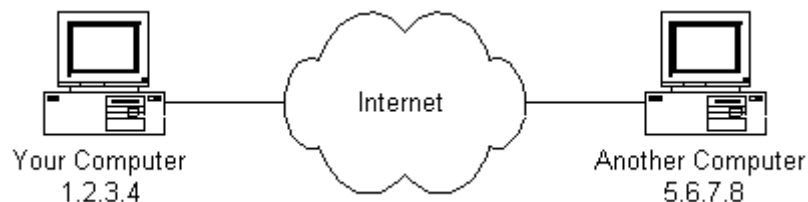


图1

如果你通过互联网服务提供商(ISP)[比如：移动、联通、电信...]连接到互联网，它们通常会为你的拨号会话期间分配一个临时IP地址，如果你从局域网(LAN)连接到互联网，你的计算机可能有一个永久IP，或者它可能从DHCP(动态主机配置协议)服务器获得一个临时的IP地址。任何情况下，如果你连接到互联网，你的计算机有一个唯一的IP地址。

|:-----|

如果你使用的是Windows或者Mac并且连接到互联网，那么有一个方便的程序可以查看互联网上的计算机是否在运行。称为 `ping`，ping可能来自旧潜艇声纳系统发出的声音。请启动命令提示符窗口：输入 `ping www.baidu.com`。ping程序将向指定的计算机发送“ping”（实际上是ICMP(互联网控制消息协议)回显请求消息）。被ping的计算机将回复。ping程序将计算过期时间，直到回复返回（如果确实如此）。此外，如果你将如域名（即www.baidu.com）而不是IP地址，ping将解析域名并显示计算机的IP地址。

```
C:\Users\...>ping www.baidu.com

正在 Ping www.a.shifen.com [180.101.49.11] 具有 32 字节的数据:
来自 180.101.49.11 的回复: 字节=32 时间=9ms TTL=50
来自 180.101.49.11 的回复: 字节=32 时间=11ms TTL=50
来自 180.101.49.11 的回复: 字节=32 时间=9ms TTL=50
来自 180.101.49.11 的回复: 字节=32 时间=9ms TTL=50

180.101.49.11 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 9ms, 最长 = 11ms, 平均 = 9ms

C:\Users\...>
```

图2

3. 协议堆栈和数据包

所以你的电脑连接上互联网上，并且有一个唯一的IP地址。它如何与其它连接到互联网上的计算机“对话”？这有个示例：假设你的IP地址是1.2.3.4，你想要向计算机5.6.7.8发动一条消息。你需要发送的消息是“Hello computer 5.6.7.8!”。显然，信息必须通过任何将你的计算机连接到互联网的线路来传输。假如你从家里拨打了ISP，那消息必须通过电话线传输。因此，必须将消息从字母文本翻译成电子信号，通过互联网传输，然后再翻译会字母文本。这是如何实现的？通过使用协议堆栈。每台计算机都需要一台在互联网上进行通信，并且它通常内置于计算机的操作系统（即Windows、Linux等）中。由于使用了两种主要的通讯协议，因此互联网上使用的协议堆栈被称为 `TCP/IP` 协议栈。TCP/IP堆栈如下所示：

协议层 | 注释

|:-----|:-----

应用协议层 | 特定于应用程序的协议，例如www、电子邮件、FTP等

传输控制协议层 | TCP使用端口号将数据包定向到计算机上的特定应用程序

网络协议层 | IP使用IP地址将数据包定向到特定的计算机

硬件层 | 将二进制数据包数据转换为网络信号并返回（例如：以太网卡、电话调制解调器等）

如果我们遵循消失“Hello computer 5.6.7.8!”的路径从我们的计算机到IP地址为5.6.7.8的计算机，它会发生这样的事情：

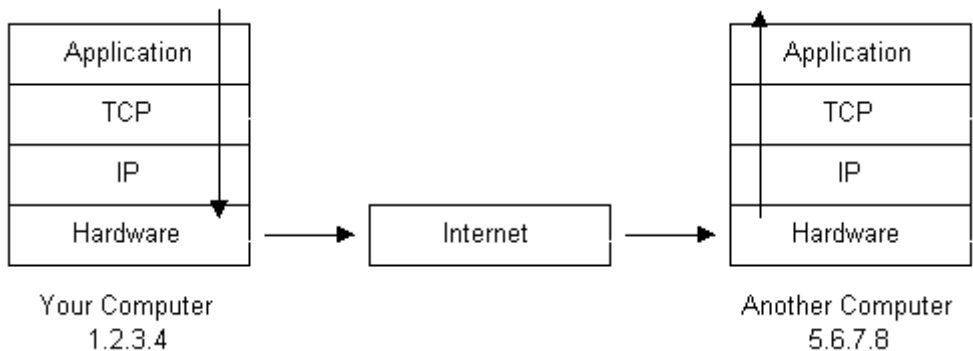


图3

1. 该消息将从你计算机上协议退栈的顶部开始，并向下工作。

2. 如果要发送的消息很长，则消息通过的每个退栈层都可能将消息分解成更小的数据块。这是因为通过互联网（和大多数计算机网络）发动的数据是以可管理的块的形式发送的。在互联网上，这些数据块成为 **数据包**。
3. 数据包将通过应用层并继续到TCP层。每个数据包都分配有一个 **端口号**。端口之后解释，但足以说明许多程序可能使用TCP/IP堆栈并发送消息。我们需要直到目标计算机上的哪个程序需要接收消息，因为它将侦听到特定端口。
4. 通过TCP层后，数据包进入IP层。这是每个数据包接收其目标地址5.6.7.8的地方。
5. 既然我们的消息包有一个端口号和一个IP地址，它们就可以通过Internet发送了。硬件层负责将包含我们消息的字母文字的数据包转换为电子信号并通过电话线（网线）传输它们。
6. 在电话线（网线）的另一端，你的ISP(互联网服务提供商)直接连接到互联网。ISP路由器（对就是家里的路由器或者光猫）检查每个数据包中的目标地址并确定将其发送到何处。通常，数据包的下站是另一个路由器。
7. 最终，数据包到达计算机5.6.7.8。在这里，数据包从目标计算机的TCP/IP堆栈的底部开始向上工作。
8. 随着数据包向上通过堆栈，发送计算机的堆栈添加的所有路由数据（例如IP地址和端口号）都从数据包中剥离。
9. 当数据包到达退栈顶时，数据包被重新组合成原来的形式，“Hello computer 5.6.7.8! ”。

4. 网络基础设施

所以现在您知道了数据包是如何通过 互联网 从一台计算机传输到另一台计算机的。但介于两者之间的是什么？究竟是什么构成了互联网？我们再看另外一张图：

这是老式的连接方式，但原理是一样的

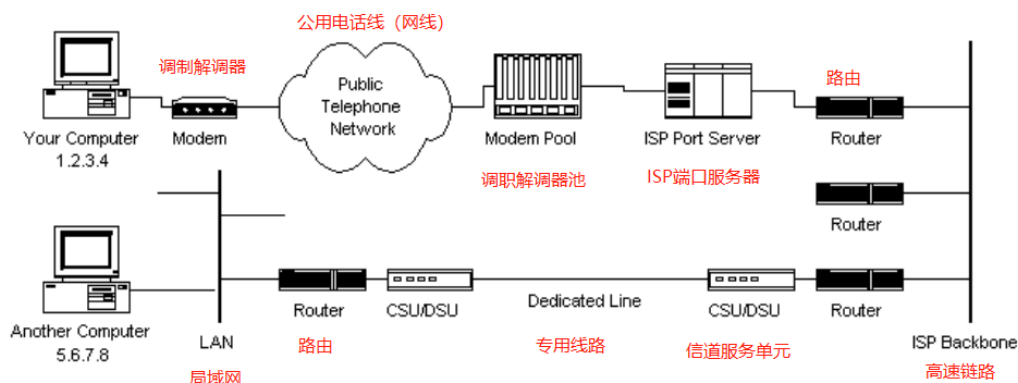


图4

这张图比2.的图更详细一些。通过公共电话线（网线）与ISP(互联网服务提供商)的物理连接可能很容易猜到，但除此之外可能需要一些解释。

ISP为他们的拨号接入客户维护一个调制解调器池。这是由某种形式的计算机（通常是专用计算机）来管理的，它控制数据从调制解调器池流向骨干网络或专用线路路由器。这个设置被称为端口服务器，因为它“服务”网络访问。计费和使用信息通常也在这里收集。

在你的数据包通过电话网络和ISP的本地设备后，它们被路由到ISP的主干网或ISP购买宽带的主干网。从这里开始，数据包通常会经过多个路由器和多个骨干网、专线和其它网络，直到找到它们的目的地，即地址为5.6.7.8的计算机。但是，如果我们知道我们的数据包在互联网上的确切路由，那不是很好吗？事实证明，有一种方法...

查看路由 - Tracertoute(路由跟踪)程序 |

|:-----|

如果你正在使用Windows或者Linux，并连接到互联网，这里有一个方便的互联网程序查看路由。程序称为 **tracrouter**（路由跟踪），它显示数据包通向指定Internet目的地的路径。像 ping一样，你必须从命令提示符窗口使用tracroute。windows和Linux下使用 **tracroute www.baidu.com**，与ping一样，您也可以输入IP地址而不是域名。Traceroute会打印出所有路由器、计算机和任何其他互联网实体的列表，你的数据包必须经过这些实体才能到达目的地。

```
C:\Users\r...>tracert www.baidu.com
```

```
通过最多 30 个跃点跟踪  
到 www.a.shifen.com [180.101.49.11] 的路由:
```

1	*	*	*	请求超时。
2	3 ms	4 ms	3 ms	1.1.1.252
3	1 ms	<1 毫秒	1 ms	10.100.2.254
4	4 ms	2 ms	3 ms	101.95.162.241
5	3 ms	3 ms	3 ms	124.74.49.185
6	3 ms	3 ms	2 ms	101.95.89.38
7	*	*	*	请求超时。
8	*	3 ms	*	59.43.80.145
9	*	*	*	请求超时。
10	*	*	*	请求超时。
11	*	10 ms	*	58.213.94.90
12	20 ms	9 ms	9 ms	58.213.96.118
13	*	*	*	请求超时。
14	*	*	*	请求超时。
15	*	*	*	请求超时。
16	10 ms	9 ms	9 ms	180.101.49.11

跟踪完成。

如果您使用tracroute，您将注意到您的数据包必须通过许多东西才能到达目的地。大多数都有很长的名字，比如sjc2-core1-h2-0-0.atlas.digex.net和fddi0-0.br4.SJC.globalcenter.net。这些是互联网路由器，可以决定把你的数据包发送到哪里。图4中显示了几个路由器，但只有少数。图4显示了一个简单的网络结构。互联网要复杂得多。

5. 互联网基础设施

互联网骨干网由许多相互连接的大型网络组成。这些大型网络被称为 **网络服务提供商** 或 **NSP**。一些大型 NSP 是 UUNet、CerfNet、IBM、BBN Planet、SprintNet、PSINet 等。这些网络相互对等以交换数据包流量。每个 NSP 都需要连接到三个网络接入点或 NAP。在 NAP 处，数据包流量可能会从一个 NSP 的主干跳转到另一个 NSP 的主干。NSP 还在 **都会区交易所** 或 **MAE** 处相互连接。MAE 的用途与 NAP 相同，但属于私人所有。NAP 是最初的 Internet 互连点。NAP 和 MAE 都称为 Internet 交换点或 **IX**。NSP 还将带宽出售给较小的网络，例如 ISP 和较小的带宽提供商。下图显示了这种分层基础设施。

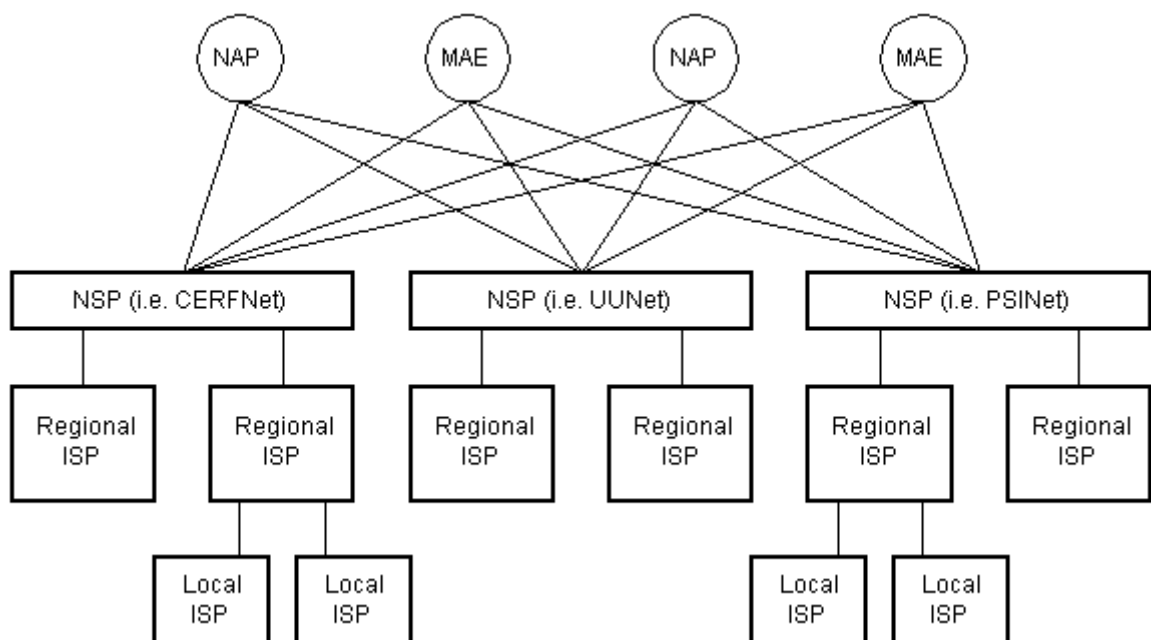


图5

这不是互联网实际部分的真实表示。图 5 仅用于演示 NSP 如何相互连接以及更小的 ISP。图 5 中没有像图 4 那样显示任何物理网络组件。这是因为单个 NSP 的骨干基础设施本身就是一个复杂的图形。大多数 NSP 在其网站上发布其网络基础设施的地图，并且很容易找到。由于互联网的规模、复杂性和不断变化的结构，要绘制一张实际的互联网地图几乎是不可能的。

6. 互联网路由层次结构

那么数据包是如何通过 Internet 找到自己的方式的呢？是否每台连接到 Internet 的计算机都知道其他计算机在哪里？数据包是否简单地“广播”到 Internet 上的每台计算机？前面两个问题的答案都是“不”。没有计算机知道任何其他计算机在哪里，并且数据包不会发送到每台计算机。用于将数据包发送到目的地的信息包含在连接到 Internet 的每个路由器保存的路由表中。

路由器是分组交换机。路由器通常连接在网络之间以在它们之间路由数据包。每个路由器都知道它的子网以及它们使用的 IP 地址。路由器通常不知道它“上方”的 IP 地址。检查下面的图 5。连接主干的黑框子是路由器。顶部较大的 NSP 主干通过 NAP 连接。在它们之下是几个子网络，在它们之下是更多的子网络。底部是两个连接有计算机的局域网。

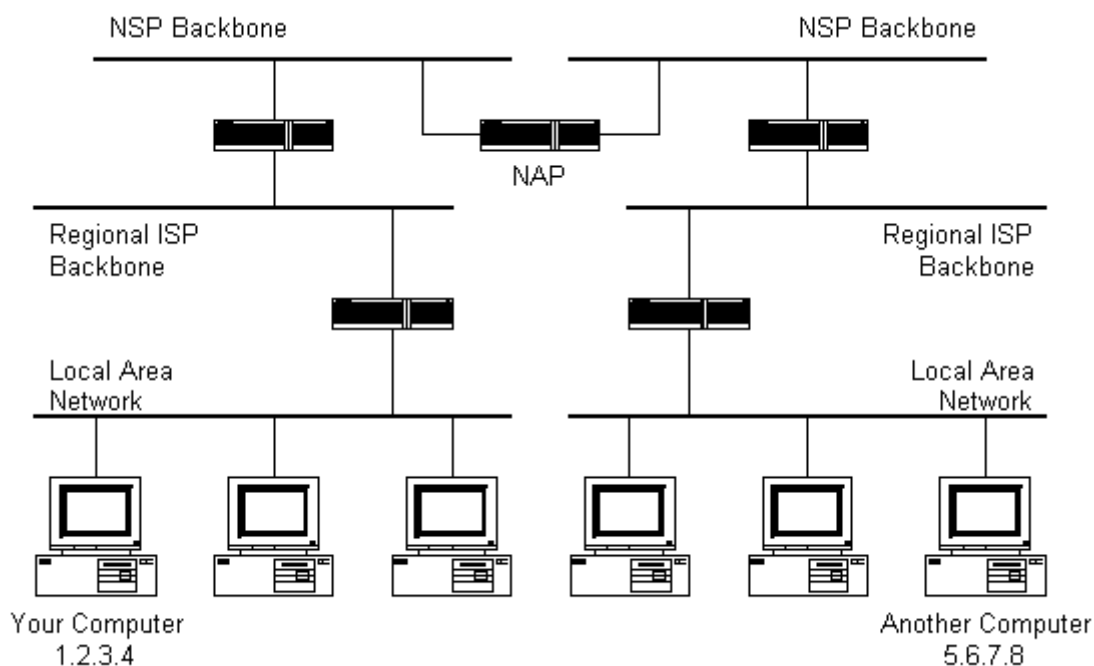


图6

当数据包到达路由器时，路由器会检查由始发计算机上的 IP 协议层放置在那里的 IP 地址。路由器检查它的路由表。如果找到包含 IP 地址的网络，则将数据包发送到该网络。如果未找到包含 IP 地址的网络，则路由器通过默认路由发送数据包，通常沿着骨干层级向上到下一个路由器。希望下一个路由器知道将数据包发送到哪里。如果没有，则数据包再次向上路由，直到到达 NSP 骨干网。连接到 NSP 骨干网的路由器拥有最大的路由表，在这里数据包将被路由到正确的骨干网，在那里它将开始“向下”通过越来越小的网络，直到找到它的目的地。

7.域名和地址解析

但是，如果你不知道要连接到的计算机的IP地址怎么办？如果您需要访问一个称为www.anothercomputer.com的web服务器呢？你的网页浏览器如何知道这台电脑在互联网上的位置？所有这些问题的答案就是域名服务或DNS。DNS是一个分布式数据库，用来记录互联网上计算机的名称和它们对应的IP地址。

许多连接到Internet的计算机主机DNS数据库的一部分和允许其他人访问它的软件。这些计算机被称为DNS服务器。没有DNS服务器包含整个数据库；它们只包含它的一个子集。如果一个DNS服务器不包含其他计算机请求的域名，DNS服务器会将请求计算机重定向到另一个DNS服务器。

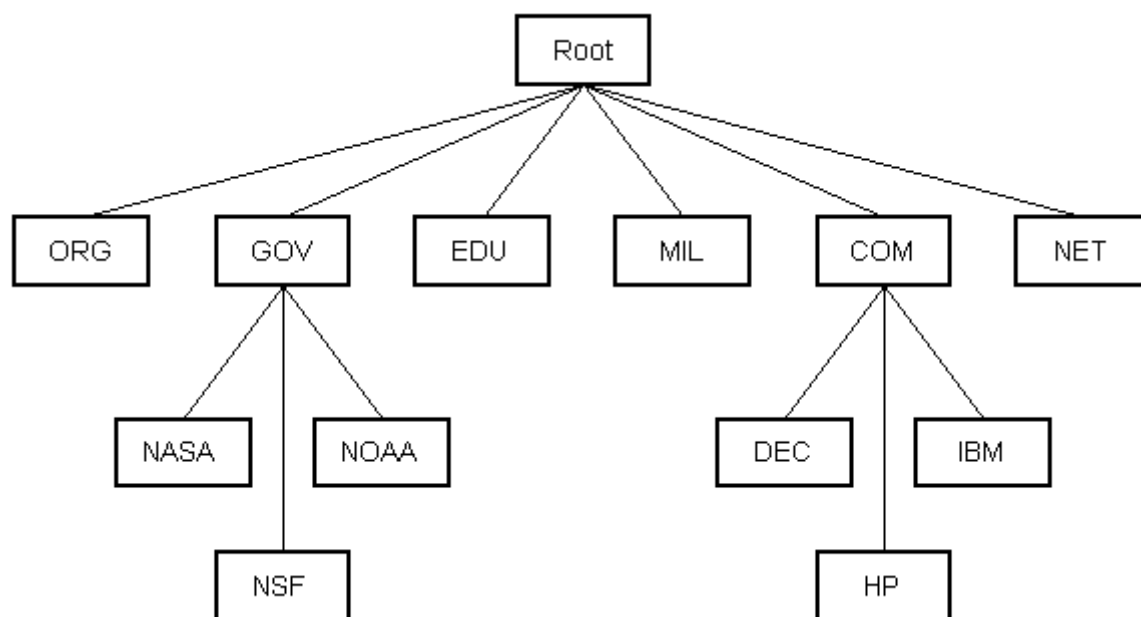


图7

域名服务的结构类似于IP路由结构。请求域名解析的计算机将被重定向到“向上”的层次结构，直到找到一个DNS服务器可以解析请求中的域名。图6说明了层次结构的一部分。在树的顶部是域根。一些更老、更常见的域名在顶部附近。没有显示的是世界各地的众多DNS服务器，它们构成了这个层次结构的其余部分。

当建立互联网连接时(例如，在Windows下的局域网或拨号网络)，通常会指定一个主要和一个或多个次要DNS服务器作为安装的一部分。这样，任何需要域名解析的Internet应用程序都能正常工作。例如，当你在浏览器中输入一个网址时，浏览器首先连接到你的首选DNS服务器。在获得您输入的域名的IP地址后，浏览器就会连接到目标计算机并请求您想要的网页。

查看域名 - 在Windows中禁用DNS |

|:-----|

如果您使用 Windows 95/NT 并访问 Internet，您可以查看您的 DNS 服务器，甚至禁用它们。

如果您使用拨号网络：打开拨号网络窗口（可以在 Windows 资源管理器中的 CD-ROM 驱动器和网上邻居上方找到）。右键单击您的 Internet 连接，然后单击属性。在连接属性窗口底部附近按下 TCP/IP 设置...按钮。

如果您永久连接到 Internet：右键单击网上邻居，然后单击属性。单击 TCP/IP 属性。选择顶部的 DNS 配置选项卡。

您现在应该查看您的 DNS 服务器的 IP 地址。在这里您可以禁用 DNS 或将您的 DNS 服务器设置为 0.0.0.0。（首先记下您的 DNS 服务器的 IP 地址。您可能还必须重新启动 Windows。）现在在您的

Web 浏览器中输入一个地址。浏览器将无法解析域名，您可能会看到一个令人讨厌的对话框，说明找不到 DNS 服务器。但是，如果您输入相应的 IP 地址而不是域名，浏览器将能够检索到所需的网页。（在禁用 DNS 之前使用 ping 获取 IP 地址。）其他 Microsoft 操作系统类似。

8. 重新审视互联网协议

正如在之前关于协议栈一节中所提到的，可以推测 Internet 上使用了許多协议。这是真的；有许多通信协议需要互联网的功能。这些协议包括 TCP 和 IP 协议、路由协议、媒体访问控制协议、应用层协议等。下面的部分描述了 Internet 上一些比较重要和常用的协议。首先讨论较高级别的协议，然后是较低级别的协议。

9. 应用协议：HTTP 和万维网

Internet 上最常用的服务之一是万维网 (WWW)。使 Web 工作的应用程序协议是超文本传输协议 或 HTTP。不要将此与超文本标记语言 (HTML) 混淆。HTML 是用于编写网页的语言。HTTP 是 Web 浏览器和 Web 服务器用于通过 Internet 相互通信的协议。它是一种应用程序级协议，因为它位于协议栈中的 TCP 层之上，并被特定应用程序用来相互通信。在这种情况下，应用程序是 Web 浏览器和 Web 服务器。

HTTP 是一种基于文本的无连接协议。客户端 (web 浏览器) 向 web 服务器发送请求，获取 web 元素，如网页和图像。当客户端和服务端通过 Internet 进行服务后，客户端和服务端之间的连接会断开。必须为每个请求建立一个新的连接。大多数协议都是面向连接的。这意味着两台互相通信的计算机在 Internet 上保持连接开放。但是，HTTP 不需要。在客户端发出 HTTP 请求之前，必须与服务器建立新的连接。

当你在 web 浏览器中键入 URL 时，会发生一下情况：

- 如果 URL 包含域名，则浏览器首先连接到域名服务器并为 Web 服务器检索相应的 IP 地址。
 - Web 浏览器连接到 Web 服务器并为所需网页发送 HTTP 请求（通过协议栈）。
 - Web 服务器接收请求并检查所需的页面。如果该页面存在，则 Web 服务器将其发送。如果服务器找不到请求的页面，它将发送 HTTP 404 错误消息。（404 表示“未找到页面”，任何上网过的人都可能知道。）
 - Web 浏览器收到返回的页面并关闭连接。
 - 然后浏览器解析页面并寻找完成网页所需的其他页面元素。这些通常包括图像、小程序等。
 - 对于需要的每个元素，浏览器都会为每个元素向服务器建立额外的连接和 HTTP 请求。
- 当浏览器完成加载所有图像、小程序等时，页面将完全加载到浏览器窗口中。

(PS：这个知识点自己没去了解，操作也没什么反应，之后学习了专业知识再来了解，太难了)

查看 HTTP - 使用 Telnet 客户端使用 HTTP 检索 Web 页面 |

|:-----|

Telnet 是 Internet 上使用的远程终端服务。它的使用最近有所下降，但它是研究 Internet 的一个非常有用的工具。在 Windows 中找到默认的 telnet 程序。它可能位于名为 telnet.exe 的 Windows 目录中。打开后，下拉终端菜单并选择首选项。在首选项窗口中，选中本地回声。（这样您可以在键入 HTTP 请求时看到它。）现在下拉连接菜单并选择远程系统。输入 www.google.com 作为主机名，输入 80 作为端口。（默认情况下，Web 服务器通常侦听端口 80。）按 Connect。现在输入 GET / HTTP/1.0，按“Enter”两次。这是一个简单的 HTTP 请求到一个 web 服务器的根页面。你应该会看到一个网页闪过，然后弹出一个对话框，告诉你连接丢失了。如果要保存检索到的页面，请在 Telnet 程序中打开登录。然后你可以浏览网页，看到用来编写它的 HTML。

大多数 Internet 协议都由称为 Request For Comments 或 RFC 的 Internet 文档指定。可以在 Internet 上的多个位置找到 RFC。有关适当的 URL，请参阅下面的参考资料部分。HTTP 版本 1.0 由 RFC 1945 指定。

10. 应用协议：SMTP和电子邮件

另一种常用的 Internet 服务是电子邮件。电子邮件使用称为简单邮件传输协议或 SMTP 的应用程序级协议。SMTP 也是基于文本的协议，但与 HTTP 不同，SMTP 是面向连接的。SMTP 也比 HTTP 更复杂。SMTP 中的命令和注意事项比 HTTP 中的要多得多。

当您打开邮件客户端阅读电子邮件时，通常会发生以下情况：

1. 邮件客户端（Netscape Mail、Gmail、Microsoft Outlook 等）打开到其默认邮件服务器的连接。邮件服务器的 IP 地址或域名通常在安装邮件客户端时设置。
2. 邮件服务器将始终发送第一条消息以标识自己。
3. 客户端将发送一个 SMTP HELO 命令，服务器将以 250 OK 消息响应该命令。
4. 根据客户端是否正在检查邮件、发送邮件等，适当的 SMTP 命令将被发送到服务器，服务器将做出相应的响应。
5. 此请求/响应事务将继续，直到客户端发送 SMTP QUIT 命令。然后服务器会说再见并关闭连接。

下面是一个简单的SMTP客户端和SMTP服务器之间的“对话”。R:服务器端(接收方)发送的消息，S:客户端(发送方)发送的消息。

这个 SMTP 示例显示了 Smith 在主机 USC-ISIF 上发送给主机 BBN-UNIX 上的 Jones、Green 和 Brown 的邮件。这里我们假设主机 USC-ISIF 直接联系主机 BBN-UNIX。琼斯和布朗接受了邮件。Green 在主机 BBN-UNIX 上没有邮箱。

```
-----  
  
R: 220 BBN-UNIX.ARPA Simple Mail Transfer Service Ready  
S: HELO USC-ISIF.ARPA  
R: 250 BBN-UNIX.ARPA  
  
S: MAIL FROM:<Smith@USC-ISIF.ARPA>  
R: 250 OK  
  
S: RCPT TO:<Jones@BBN-UNIX.ARPA>  
R: 250 OK  
  
S: RCPT TO:<Green@BBN-UNIX.ARPA>  
R: 550 No such user here  
  
S: RCPT TO:<Brown@BBN-UNIX.ARPA>  
R: 250 OK  
  
S: DATA  
R: 354 Start mail input; end with <CRLF>.<CRLF>  
S: Blah blah blah...  
S: ...etc. etc. etc.  
S: .  
R: 250 OK  
  
S: QUIT  
R: 221 BBN-UNIX.ARPA Service closing transmission channel
```

此SMTP事务取自指定SMTP的RFC 821。

11. 传输控制协议

在协议栈的应用层之下是 TCP 层。当应用程序打开与 Internet 上另一台计算机的连接时，它们发送的消息（使用特定的应用程序层协议）通过堆栈向下传递到 TCP 层。TCP 负责将应用程序协议路由到目标计算机上的正确应用程序。为此，使用端口号。端口可以被认为是每台计算机上的独立通道。例如，您可以一边浏览网页一边阅读电子邮件。这是因为这两个应用程序（Web 浏览器和邮件客户端）使用了不同的端口号。当数据包到达计算机并沿协议栈向上移动时，TCP 层会根据端口号决定哪个应用程序接收数据包。

TCP是这样工作的：

- 当TCP层从上面接收到应用层协议数据时，它将其分割成可管理的“块”，然后添加一个带有特定TCP信息的TCP报头到每个“块”。TCP报头中包含的信息包括数据需要发送到的应用程序的端口号。
- 当TCP层从它下面的IP层接收到一个包时，TCP层从包中取出TCP报头数据，如果需要进行一些数据重构，然后使用从TCP报头中获取的端口号将数据发送给正确的应用程序。

这就是 TCP 如何将通过协议栈的数据路由到正确的应用程序。

TCP 不是文本协议。TCP 是一种面向连接的、可靠的字节流服务。面向连接意味着使用 TCP 的两个应用程序必须首先建立连接，然后才能交换数据。TCP 是可靠的，因为对于接收到的每个数据包，都会向发送方发送确认以确认传递。TCP 还在其标头中包含一个校验和，用于对接收到的数据进行错误检查。

TCP 标头如下所示：

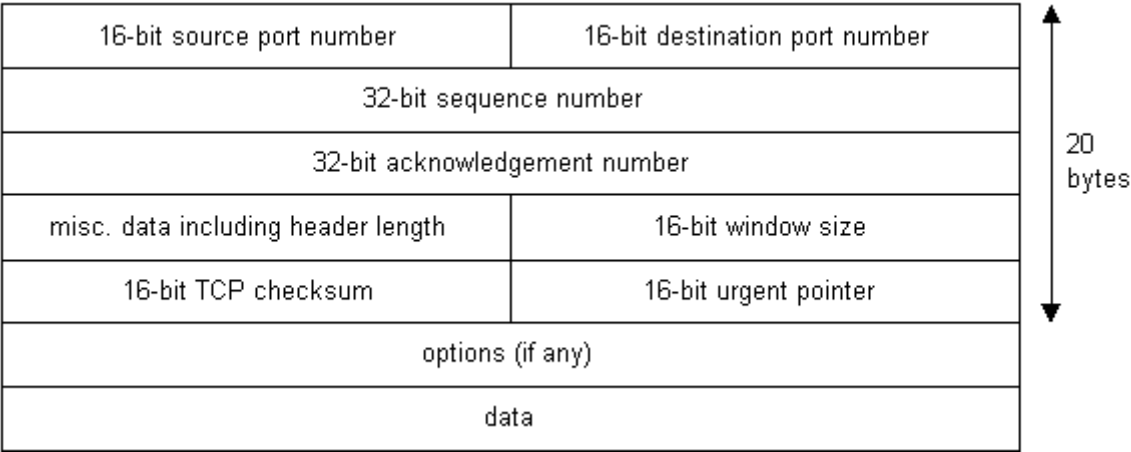


图8

注意，在TCP报头中没有IP地址的位置。这是因为TCP对IP地址一无所知。TCP的工作是可靠地从一个应用程序获取到另一个应用程序的应用层数据。将数据从一台计算机传输到另一台计算机是IP的工作。

查看端口- 默认的互联网端口数字 |

|:-----:|

下面列出的是一些更常用的互联网服务的端口号

FTP 20/21

Telnet 23

SMTP 25

HTTP 80

Quake III Arena 27960

12. 互联网协议

与 TCP 不同，IP 是一种不可靠的无连接协议。IP 不关心数据包是否到达目的地。IP 也不知道连接和端口号。IP 的工作也是向其他计算机发送和路由数据包。IP 数据包是独立的实体，可能无序到达或根本没有到达。确保数据包到达并按正确顺序是 TCP 的工作。IP 与 TCP 唯一的共同点是它接收数据并将其自己的 IP 标头信息添加到 TCP 数据的方式。IP 标头如下所示：

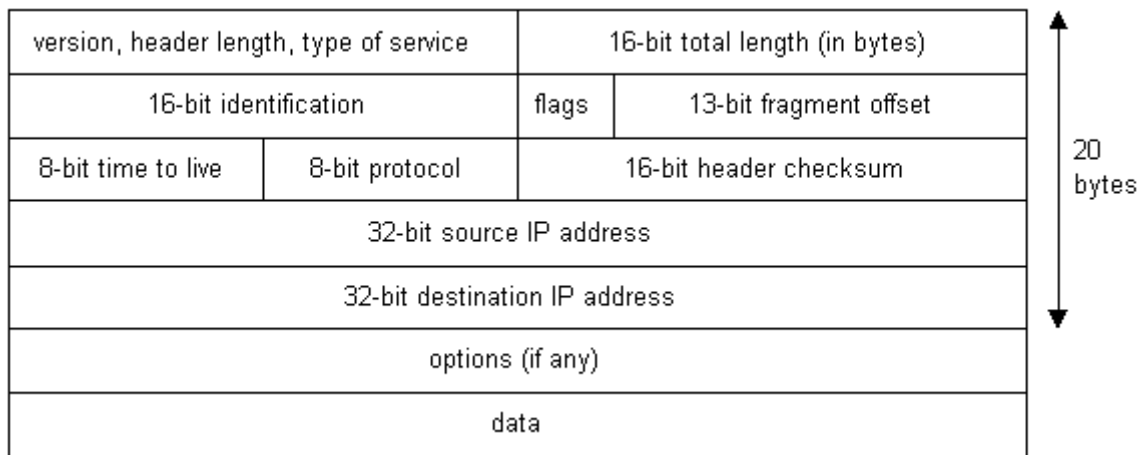


图9

上面我们在IP头中看到了发送和接收计算机的IP地址。下面是数据包经过应用层、TCP层和IP层后的样子。应用层数据在TCP层进行分段，添加TCP报头，报文继续到IP层，添加IP报头，然后报文在Internet上传输。

一段完整的数据包：

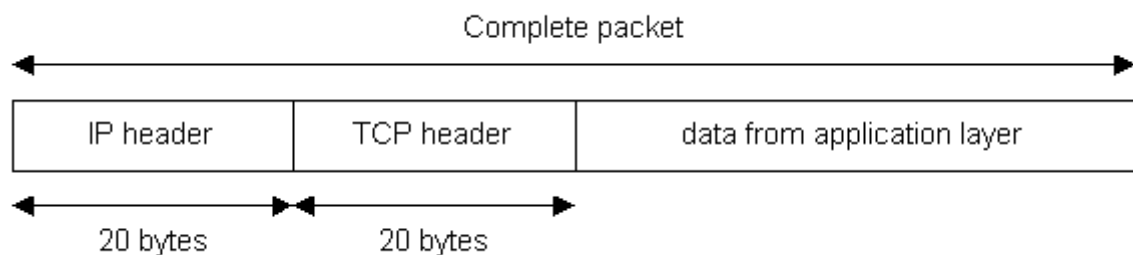


图10

13. 总结

现在您知道 Internet 是如何工作的了。但这种情况会持续多久？目前互联网上使用的IP版本（版本4）只允许232个地址。最终不会有任何空闲的 IP 地址。惊讶吗？别担心。IP 版本 6 目前正在由研究机构和公司组成的联盟在研究骨干上进行测试。在那之后？谁知道。自从作为国防部研究项目成立以来，互联网已经走过了漫长的道路。没有人真正知道互联网会变成什么样子。然而，有一件事是肯定的。互联网将以前所未有的方式将世界团结起来。信息时代正在大踏步前进，我很高兴能成为其中的一员。

Rus Shuler, 1998

2002 年更新

PS:这篇文章有一定的年代感了，但是其中的知识点和原理还是和现在一样的，即使现在互联网爆炸式发展，但底层的思想和逻辑还是不会变的。这也是为什么我很痛苦的把这篇文章看完了，虽然有很多地方都不懂，但是慢慢来...

14. 资源

文章列举的很多资源都不能打开了，或者已经不维护了。只有一个比较贴合现在的使用场景

<http://www.internetweather.com/> 显示了互联网延迟的动画地图。

15. 图书

都是一些90年代的书籍，而且是英文书，太难了，先放弃了，之后有时间在研究（估计也没时间了:））

(<http://web.stanford.edu/class/msande91si/www-spr04/readings/week1/InternetWhitepaper.htm>)

