

理解。例如，如果一个服务器写一个已经被客户端关闭了的连接(比如，因为你在浏览器上单击了“Stop”按钮)，那么第一次这样的写会正常返回，但是第二次写就会引起发送 SIGPIPE 信号，这个信号的默认行为就是终止这个进程。如果捕获或者忽略 SIGPIPE 信号，那么第二次写操作会返回值 -1，并将 `errno` 设置为 `EPIPE`。`strerr` 和 `perror` 函数将 `EPIPE` 错误报告为“Broken pipe”，这是一个迷惑了很多人的不太直观的信息。总的来说，一个健壮的服务端必须捕获这些 SIGPIPE 信号，并且检查 `write` 函数调用是否有 `EPIPE` 错误。

## 11.7 小结

每个网络应用都是基于客户端-服务器模型的。根据这个模型，一个应用是由一个服务器和一个或多个客户端组成的。服务器管理资源，以某种方式操作资源，为它的客户端提供服务。客户端-服务器模型中的基本操作是客户端-服务器事务，它是由客户端请求和跟随其后的服务器响应组成的。

客户端和服务端通过因特网这个全球网络来通信。从程序员的角度来看，我们可以把因特网看成是一个全球范围的主机集合，具有以下几个属性：1)每个因特网主机都有一个唯一的 32 位名字，称为它的 IP 地址。2)IP 地址的集合被映射为一个因特网域名的集合。3)不同因特网主机上的进程能够通过连接互相通信。

客户端和服务端通过使用套接字接口建立连接。一个套接字是连接的一个端点，连接以文件描述符的形式提供给应用程序。套接字接口提供了打开和关闭套接字描述符的函数。客户端和服务端通过读写这些描述符来实现彼此间的通信。

Web 服务器使用 HTTP 协议和它们的客户端(例如浏览器)彼此通信。浏览器向服务器请求静态或者动态的内容。对静态内容的请求是通过从服务器磁盘取得文件并把它返回给客户端来服务的。对动态内容的请求是在服务器上一个子进程的上下文中运行一个程序并将它的输出返回给客户端来服务的。CGI 标准提供了一组规则，来管理客户端如何将程序参数传递给服务器，服务器如何将参数以及其他信息传递给子进程，以及子进程如何将它的输出发送回客户端。只用几百行 C 代码就能实现一个简单但是有功效的 Web 服务器，它既可以提供静态内容，也可以提供动态内容。

## 参考文献说明

有关因特网的官方信息源被保存在一系列的可免费获取的带编号的文档中，称为 RFC(Requests for Comments，请求注解，Internet 标准(草案))。在以下网站可获得可搜索的 RFC 的索引：

<http://rfc-editor.org>

RFC 通常是因特网基础设施的开发者编写的，因此，对于普通读者来说，往往过于详细了。然而，要想获得权威信息，没有比它更好的信息来源了。HTTP/1.1 协议记录在 RFC 2616 中。MIME 类型的权威列表存在：

<http://www.iana.org/assignments/media-types>

Kerrisk 是全面 Linux 编程的圣经，提供了现代网络编程的详细讨论[62]。关于计算机网络互联有大量很好的通用文献[65，84，114]。伟大的科技作家 W. Richard Stevens 编写了一系列相关的经典文献，如高级 Unix 编程[111]、因特网协议[109，120，107]，以及 Unix 网络编程[108，110]。认真学习 Unix 系统编程的学生会想要研究所有这些内容。不幸的是，Stevens 在 1999 年 9 月 1 日逝世。我们会永远记住他的贡献。

## 家庭作业

- \*\* 11.6 A. 修改 TINY 使得它会原样返回每个请求行和请求报头。
- B. 使用你喜欢的浏览器向 TINY 发送一个对静态内容的请求。把 TINY 的输出记录到一个文件中。
- C. 检查 TINY 的输出，确定你的浏览器使用的 HTTP 的版本。
- D. 参考 RFC 2616 中的 HTTP/1.1 标准，确定你的浏览器的 HTTP 请求中每个报头的含义。你可以从 [www.rfc-editor.org/rfc.html](http://www.rfc-editor.org/rfc.html) 获得 RFC 2616。
- \*\* 11.7 扩展 TINY，使得它可以提供 MPG 视频文件。用一个真正的浏览器来检验你的工作。