

```


1  linux> telnet kittyhawk.cmcl.cs.cmu.edu 8000  Client: open connection
2  Trying 128.2.194.242...
3  Connected to kittyhawk.cmcl.cs.cmu.edu.
4  Escape character is '^]'.
5  GET /cgi-bin/adder?15000&213 HTTP/1.0  Client: request line
6                                     Client: empty line terminates headers
7  HTTP/1.0 200 OK                      Server: response line
8  Server: Tiny Web Server                Server: identify server
9  Content-length: 115                    Adder: expect 115 bytes in response body
10 Content-type: text/html                Adder: expect HTML in response body
11                                     Adder: empty line terminates headers
12 Welcome to add.com: THE Internet addition portal. Adder: first HTML line
13 <p>The answer is: 15000 + 213 = 15213  Adder: second HTML line in response body
14 <p>Thanks for visiting!                 Adder: third HTML line in response body
15 Connection closed by foreign host.     Server: closes connection
16 linux>                                Client: closes connection and terminates

```

图 11-28 一个提供动态 HTML 内容的 HTTP 事务

旁注 将 HTTP POST 请求中的参数传递给 CGI 程序

对于 POST 请求，子进程也需要重定向标准输入到已连接描述符。然后，CGI 程序会从标准输入中读取请求主体中的参数。

 **练习题 11.5** 在 10.11 节中，我们警告过你关于在网络应用中使用 C 标准 I/O 函数的危险。然而，图 11-27 中的 CGI 程序却能没有任何问题地使用标准 I/O。为什么呢？

11.6 综合：TINY Web 服务器

我们通过开发一个虽小但功能齐全的称为 TINY 的 Web 服务器来结束对网络编程的讨论。TINY 是一个有趣的程序。在短短 250 行代码中，它结合了许多我们已经学习到的思想，例如进程控制、Unix I/O、套接字接口和 HTTP。虽然它缺乏一个实际服务器所具备的功能性、健壮性和安全性，但是它足够用来为实际的 Web 浏览器提供静态和动态的内容。我们鼓励你研究它，并且自己实现它。将一个实际的浏览器指向你自己的服务器，看着它显示一个复杂的带有文本和图片的 Web 页面，真是非常令人兴奋(甚至对我们这些作者来说，也是如此!)

1. TINY 的 main 程序

图 11-29 展示了 TINY 的主程序。TINY 是一个迭代服务器，监听在命令行中传递来的端口上的连接请求。在通过调用 `open_listenfd` 函数打开一个监听套接字以后，TINY 执行典型的无限服务器循环，不断地接受连接请求(第 32 行)，执行事务(第 36 行)，并关闭连接的它那一端(第 37 行)。

2. doit 函数

图 11-30 中的 `doit` 函数处理一个 HTTP 事务。首先，我们读和解析请求行(第 11~14 行)。注意，我们使用图 11-8 中的 `rio_readlineb` 函数读取请求行。

TINY 只支持 GET 方法。如果客户端请求其他方法(比如 POST)，我们发送给它一个错误信息，并返回到主程序(第 15~19 行)，主程序随后关闭连接并等待下一个连接请求。否则，我们读并且(像我们将要看到的那样)忽略任何请求报头(第 20 行)。