



图 3-9 PUT 示例

PUT 方法的语义就是让服务器用请求的主体部分来创建一个由所请求的 URL 命名的新文档，或者，如果那个 URL 已经存在的话，就用这个主体来替代它。

因为 PUT 允许用户对内容进行修改，所以很多 Web 服务器都要求在执行 PUT 之前，用密码登录。在第 12 章中可以读到更多有关密码认证的内容。

### 3.3.5 POST

POST 方法起初是用来向服务器输入数据的<sup>3</sup>。实际上，通常会用它来支持 HTML 的表单。表单中填好的数据通常会被送给服务器，然后由服务器将其发送到它要去的地方（比如，送到一个服务器网关程序中，然后由这个程序对其进行处理）。图 3-10 显示了一个用 POST 方法发起 HTTP 请求——向服务器发送表单数据——的客户端。

### 3.3.6 TRACE

客户端发起一个请求时，这个请求可能要穿过防火墙、代理、网关或其他一些应用程序。每个中间节点都可能会修改原始的 HTTP 请求。TRACE 方法允许客户端在最终将请求发送给服务器时，看看它变成了什么样子。

TRACE 请求会在目的服务器端发起一个“环回”诊断。行程最后一站的服务器会弹回一条 TRACE 响应，并在响应主体中携带它收到的原始请求报文。这样客户端就可以查看在所有中间 HTTP 应用程序组成的请求/响应链上，原始报文是否，以及如何被毁坏或修改过（参见图 3-11）。

55

注 3: POST 用于向服务器发送数据。PUT 用于向服务器上的资源（例如文件）中存储数据。