有数百万的人在用 Web 进行私人事务处理，访问私有的数据。通过 Web 可以很方便地访问这些信息，但仅仅是方便访问还是不够的。我们要保证只有特定的人能看到我们的敏感信息并且能够执行我们的特权事务。并不是所有的信息都能够公开发布的。

未授权用户无法查看我们的在线旅游档案，也不能在未经许可的情况下向 Web 站点发布文档，这会让我们感觉舒服一些。我们还要确保，组织中未经授权或不怀好意的成员无法获取那些最敏感的公司计划文档。我们与孩子、配偶以及暗恋对象的私人 Web 通信都是在带有些许隐私保护的情况下进行的，这样我们才能放心。

服务器需要通过某种方式来了解用户身份。一旦服务器知道了用户身份，就可以判定用户可以访问的事务和资源了。认证就意味着要证明你是谁。通常是通过提供用户名和密码来进行认证的。HTTP 为认证提供了一种原生工具。尽管我们可以在 HTTP 的认证形式和 cookie 基础之上“运行自己的”认证工具，但在很多情况下，HTTP 的原生认证功能就可以很好地满足要求。

本章阐述了 HTTP 的认证机制，深入介绍了最常见的 HTTP 认证形式，*基本认证*（basic authentication）。下一章将介绍一种称为*摘要认证*（digest authentication）的功能更强的认证技术。

12.1 认证

认证就是要给出一些身份证明。当出示像护照或驾照那样有照片的身份证件时，就给出了一些证据，说明你就是你所声称的那个人。在自动取款机上输入 PIN 码，或
277 在计算机系统的对话框中输入了密码时，也是在证明你就是你所声称的那个人。

现在，这些策略都不是绝对有效的。密码可以被猜出来或被人偶然听到，身份证件可能被偷去或被伪造，但每种证据都有助于构建合理的信任，说明你就是你所声称的那个人。

12.1.1 HTTP的质询/响应认证框架

HTTP 提供了一个原生的*质询 / 响应*（challenge/response）框架，简化了对用户的认证过程。HTTP 的认证模型如图 12-1 中所示。

Web 应用程序收到一条 HTTP 请求报文时，服务器没有按照请求执行动作，而是以一个“认证质询”进行响应，要求用户提供一些保密信息来说明他是谁，从而对其进行质询。

用户再次发起请求时，要附上保密证书（用户名和密码）。如果证书不匹配，服务器可以再次质询客户端，或产生一条错误信息。如果证书匹配，就可以正常完成请求了。