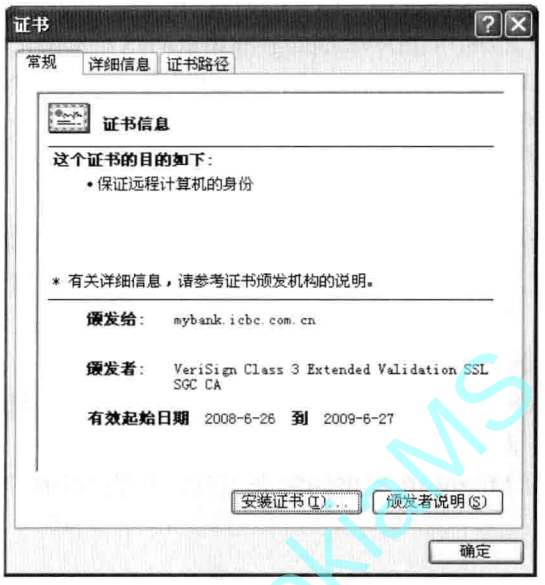

图 21-1　网站证书的内容

用户需要仔细核对该证书中网站信息是否正确，如发现疑问或异常，则该网站可能是假网站。网站信息主要包括：国家 C、省份 S、城市 L、单位 U、部门 OU、名称 CN 等。

21.2　防止假软件与代码签名证书

本节以网上银行为例来说明如何防止假软件。

21.2.1　Web 技术的发展

由于 Web 技术具有跨平台和瘦客户端等技术优势，因此网上银行及电子交易类系统普遍采用 Web 技术，既简化了用户的操作难度，又降低了系统的运维成本。

Web 技术主要由传输协议、服务器端技术和浏览器端技术组成。

最早的 Web 技术起源于静态页面的浏览，传输协议采用 HTTP，服务器端只负责存储 HTML 静态页面，浏览器端只负责显示 HTML 静态页面。为保证浏览器端计算机的安全性，浏览器端在操作系统环境中构建封闭的、安全隔离的动态运行环境，不允许访问任何本地软硬件资源（如文件、磁盘等）。

由于基于静态页面浏览的 Web 技术存在很多局限性，为适应复杂多变的 Web 应用需求，Web 技术自诞生至今的十几年内得到了快速发展。

1．传输协议

传输协议已由最初的 HTTP/1.0 发展到目前的 HTTP/1.1。

HTTP 协议是基于请求 / 响应模式的。浏览器与服务器建立连接后，发送一个请求给服务器。服务器接到请求后，进行相应处理并返回对应的响应信息。浏览器收到响应后，进行处理并显示。

HTTP/1.1 相较于 HTTP/1.0 主要区别有：缓存处理、带宽优化及网络连接的使用、错误通知的管理、消息在网络中的发送、互联网地址的维护、安全性及完整性等。

2. 服务器端技术

服务器端技术的发展目的是静态向动态发展，主要包括 CGI、PHP、ASP、ASP.NET、Servlet 和 JSP 技术。

CGI（Common Gateway Interface）技术，即公共网关接口技术。最早的 Web 服务器简单地响应浏览器发来的 HTTP 请求，并将存储在服务器上的 HTML 文件返回给浏览器。CGI 是第一种使服务器能根据运行时的具体情况，动态生成 HTML 页面的技术。1993 年，NCSA（National Center for Supercomputing Applications）提出 CGI1.0 的标准草案，之后分别在 1995 年和 1997 年制定了 CGI 1.1 和 1.2 标准。CGI 技术允许服务段的应用程序根据客户端的请求动态生成 HTML 页面，这使客户端和服务器端的动态信息交换成为了可能。随着 CGI 技术的普及，聊天室、论坛、电子商务、信息查询、全文检索等各式各样的 Web 应用蓬勃兴起，人们可以享受到信息检索、信息交换、信息处理等各种更为便捷的信息服务了。

PHP（Personal Home Page Tools）技术是 1994 年由 Rasmus Lerdorf 发明专用于 Web 服务端编程的 PHP 语言。与以往的 CGI 程序不同，PHP 语言将 HTML 代码和 PHP 指令合成为完整的服务端动态页面，Web 应用的开发者可以用一种更加简便、快捷的方式实现动态 Web 功能。

ASP（Active Server Pages）技术即活动服务器页面技术。1996 年，Microsoft 借鉴 PHP 的思想，在其 Web 服务器 IIS 3.0 中引入了 ASP 技术。ASP 使用的脚本语言是我们熟悉的 VBScript 和 Javascript。借助 Microsoft Visual Studio 等开发工具在市场上的成功，ASP 迅速成为 Windows 系统下 Web 服务端的主流开发技术。

ASP.NET 技术是面向下一代企业级网络计算的 Web 平台，是对传统 ASP 技术的重大升级和更新。ASP.NET 是建立.NET Framework 的公共语言运行库上的编程框架，可用于在服务器上生成功能强大的 Web 应用程序。

Servlet 与 JSP 技术由以 Sun 公司为首的 Java 阵营于 1997 年和 1998 年分别推出。JSP 与 Servlet 的组合让 Java 开发者同时拥有了类似 CGI 程序的集中处理功能和类似 PHP 的 HTML 嵌入功能。此外，Java 的运行时编译技术也大大提高了 Servlet 和 JSP 的执行效率。Servlet 和 JSP 被后来的 J2EE 平台吸纳为核心技术。

3. 浏览器端技术

浏览器端技术发展的目的是从静态向动态发展、从无权访问本地资源向有权访问发展，主要包括 HTML 语言、Java Applets、脚本程序、CSS、DHTML、插件技术等。其中，插件技术主要解决本地资源访问的问题。

HTML 是 Hypertext Markup Language（超文本标记语言）的缩写，它是构成 Web 页面的主要工具。

Java Applet，即 Java 小应用程序。使用 Java 语言创建小应用程序，浏览器可以将 Java Applets 从服务器下载到浏览器，在浏览器所在的机器上运行。Java Applets 可提供动画、音频和音乐等多媒体服务。1996 年，著名的 Netscape 浏览器在其 2.0 版本中率先提供了对

Java Applets 的支持，随后，Microsoft 的 IE 3.0 也在这一年开始支持 Java 技术。Java Applets 使得 Web 页面从只能展现静态的文本或图像信息，发展到可以动态展现丰富多样的信息。动态 Web 页面不仅仅表现在网页的视觉展示方式上，更重要的是它可以对网页中的内容进行控制与修改。

脚本程序是嵌入在 HTML 文档中的程序。使用脚本程序可以创建动态页面，从而大大提高了交互性。用于编写脚本程序的语言主要有 JavaScript 和 VBScript。JavaScript 由 Netscape 公司开发，具有易于使用、变量类型灵活和无须编译等特点。VBScript 由 Microsoft 公司开发，与 JavaScript 一样，可用于设计交互的 Web 页面。需要说明的是，虽然 JavaScript 和 VBScript 语言最初都是为创建客户端动态页面而设计的，但它们都可以用于服务端脚本程序的编写。客户端脚本与服务端脚本程序的区别在于执行的位置不同，前者在客户端机器执行，而后者是在 Web 服务端机器执行。

CSS（Cascading Style Sheets），即级联样式表。1996 年底，W3C 提出了 CSS 的建议标准，同年，IE 3.0 引入了对 CSS 的支持。CSS 大大提高了开发者对信息展现格式的控制能力，1997 年的 Netscape 4.0 不但支持 CSS，而且增加了许多 Netscape 公司自定义的动态 HTML 标记，这些标记在 CSS 的基础上让 HTML 页面中的各种要素“活动”了起来。

DHTML（Dynamic HTML），即动态 HTML。1997 年，微软发布了 IE 4.0，并将动态 HTML 标记、CSS 和动态对象（Dynamic Object Model）发展成为一套完整、实用、高效的客户端开发技术体系，微软称其为 DHTML。同样是实现 HTML 页面的动态效果，DHTML 技术无须启动 Java 虚拟机或其他脚本环境，可以在浏览器的支持下，获得更好的展现效果和更高的执行效率。

插件技术大大丰富了浏览器的多媒体信息展示功能，常见的插件包括 QuickTime、Realplayer、Media Player 和 Flash 等。为了在 HTML 页面中实现音频、视频等更为复杂的多媒体应用，1996 年的 Netscape 2.0 成功地引入了对 QuickTime 插件的支持，插件这种开发方式也迅速风靡了浏览器的世界。同年，在 Windows 平台上，微软将 COM 和 ActiveX 技术应用于 IE 浏览器中，其推出的 IE 3.0 正式支持在 HTML 页面中插入 ActiveX 控件，这为其他厂商扩展 Web 客户端的信息展现方式提供了方便的途径。1999 年，Realplayer 插件先后在 Netscape 和 IE 浏览器中取得了成功，与此同时，微软自己的媒体播放插件 Media Player 也被预装到了各种 Windows 版本之中。同样具有重要意义的还有 Flash 插件的问世：20 世纪 90 年代初期，Jonathan Gay 在 FutureWave 公司开发了一种名为 Future Splash Animator 的二维矢量动画展示工具，1996 年，Macromedia 公司收购了 FutureWave，并将 Jonathan Gayde 的发明改名为我们熟悉的 Flash。从此，Flash 动画成了 Web 开发者表现自我、展示个性的最佳方式。

21.2.2 插件技术与假网银软件

插件技术主要用于解决浏览器对本地资源的访问问题。网上银行和电子交易网站采用数字证书技术来保证其安全性，且用户的私钥及数字证书存储在本机密码模块中，属于本地资源。

为了在网上银行或电子交易业务操作过程中访问用户的密码模块，需要在网站中增加插件形式的网银软件或交易软件，通过该软件来实现对用户密码模块的各种密码操作和数

据访问，因此该软件的安全性直接影响到网上银行或电子交易系统的安全性。

由于Web技术是开放的，且网上银行和电子交易直接与资金有关，因此该类网站逐渐成为不法分子的攻击对象，针对该类网站的木马程序显著增加，这种木马程序伪装成网银软件或交易软件，对用户本地密码模块进行偷窃、劫持等，导致用户资金被非法转移。

21.2.3 使用代码签名证书预防假网银软件

代码签名证书为软件开发商提供了一个理想的解决方案，使得软件开发商能对其软件代码进行数字签名。通过对代码的数字签名来标识软件来源以及软件开发者的真实身份，保证代码在签名之后不被恶意篡改。使用户在下载已经签名的代码时能够有效地验证该代码的可信度。

从用户角度，可以通过代码签名服务鉴别软件的发布者及软件在传输过程中是否被篡改。如果某软件在用户计算机上执行后造成恶性后果，由于代码签名服务的可审计性，用户可依法向软件发布者索取赔偿，此举可有效制止软件开发者发布攻击性代码的行为。

从软件开发者和Web管理者的角度，利用代码签名的抗伪造性，可为其商标和产品建立一定的信誉。利用可信代码服务，一方面开发者可借助代码签名获取更高级别权限的API，设计各种功能强大的控件和桌面应用程序来创建出丰富多彩的页面；另一方面用户也可以理性地选择所需下载的软件包。并且利用代码签名技术，还可以大大减少客户端防护软件误报病毒或恶意程序的可能性，使用户在多次成功下载并运行具有代码签名的软件后，和开发者间的信任关系得到巩固。

1. 申请代码签名证书

直接向CA系统申请代码签名证书。

2. 对网银软件进行代码签名

使用微软提供的代码签名工具signcode.exe对网银软件进行代码签名。

3. 用户鉴别真假网银软件

当用户下载网银软件控件时，系统会提示控件是否具有合法的代码签名，如图21-2所示。

图21-2 控件下载安全警告

用户可以单击“代码签名”，查看该控件是由谁发布的，如图21-3所示。

单击“查看证书”按钮，可以查看发行者的代码签名证书的具体内容，如图21-4所示。

图 21-3　代码签名信息

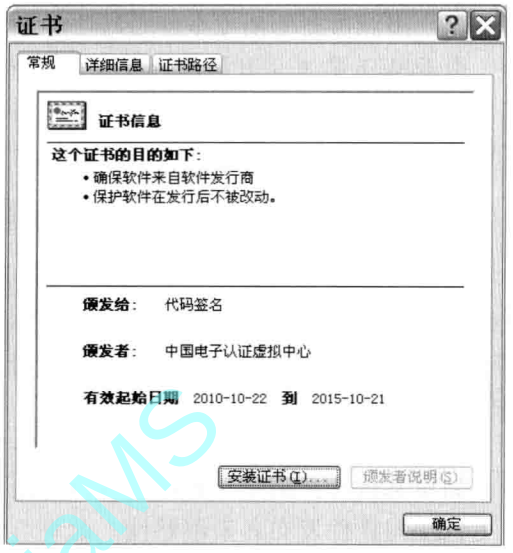

图 21-4　代码签名证书

若该控件的发行者是可信的，如商业银行等，则单击“安装”按钮即可。当出现以下情况时，则该控件可疑，建议用户不要下载：

① 该控件没有代码签名。

② 该控件的代码签名证书无效。

③ 该控件的代码签名证书内容可疑或未知。

21.3　网上银行系统

21.3.1　简介

网上银行是商业银行基于互联网为客户提供安全、实时的银行业务服务。作为一种全新的银行客户服务提交渠道，客户可以不必亲身去银行办理业务，只要能够上网，无论在家里、办公室，还是在旅途中，都能够每天 24 小时安全便捷地管理自己的资产，或者办理查询、转账、缴费等银行业务。

目前国际金融界的发展状况表明，尽管不同的银行有其不同的发展战略，目前正处在不同的发展阶段，但有一点是肯定的，即随着 Internet 的不断发展，随着金融业的不断创新，网上银行必将包含银行所有的业务，成为银行主要的业务手段。网上银行的业务主要有以下几类。

（1）查询类

查询类业务主要指账户查询。对于网上银行所实现的基本功能，各家银行对其所归置的类别和层级有所不同。账户查询主要包括账户信息查询、账户余额查询以及账户交易明细查询。出于用户对于资金全面的了解和掌控，各家银行业已将资产负债一览作为基本功能，向用户呈现活期、定期、基金、股票、贷款、信用卡等的大概情况。

（2）交易类

交易类业务主要分为转账汇款、网上支付、自助缴费几类。转账汇款基本功能包括同