基于 ASP. NET 技术的 Web 应用系统安全机制分析与设计

赵 争

(中国科学院研究生院 北京 100049)

摘 要 本文重点探讨基于 ASP. NET 技术开发 Web 应用系统的安全问题。分析了用 ASP. NET 技术开发 Web 应用系统易产生的安全隐患,对操作系统、开发平台和数据库之间的安全协作机制进行了分析,并重点讨论了数据库资源的保护问题,最后以一个文档管理系统为例,具体说明了如何实现对数据库资源的安全访问。

关键词 ASP. NET,Web 应用系统,安全,数据库,文档管理系统

Based on ASP. NET Web Applications Security Mechanism Analysis and Design

ZHAO Zheng

(Graduate University of Chinese Academy of Sciences, Beijing 100049)

Abstract In this paper, the safety of Web application based on ASP. NET technology is mainly discussed. The hidden safety trouble of Web application safety is analyzed first, as well as the cooperation mechanism between operating system, developing platform and database, with mainly discussing the protection of database resources. Then a document management system with protecting database resources is designed to explain how to access database safely.

Keywords ASP. NET, Web application system, Safety, Database, Document management system

随着 Internet 的发展,Web 技术日新月异,人们已经不再满足于静态 HTML 技术,更多的是要求动态、交互的网络技术。继 CGI、ASP、JSP、PHP 之后,微软推出的 IIS+ASP. NET 解决方案作为一种典型的服务器端网页设计技术,被广泛应用在网上银行、电子商务、BBS、搜索引擎等各种 Web 应用中。ASP. NET 是建立在公共语言运行库(CLR)上的编程框架,开发人员可充分利用整个平台的威力和灵活性。由于 ASP. NET 是面向对象的,实现了即时编译,有较好的稳定性和反馈速度,能够满足 Web 应用程序开发所要求的高效、安全、稳定和易维护等特性。与此同时,SQL Server 数据库作为微软推出的标准引擎的桌面型数据库系统,由于具有性能好、高访问量时仍可保持系统稳定等特点,具有较大的用户群体。目前,IIS+ASP. NET+SQL Server 是很多 Web 应用系统的首选开发平台。但是,该解决方案在为网络编程带来便捷的同时,也带来了严峻的安全性问题。

1 用 ASP. NET 开发 Web 系统易存在的安全隐患

IIS+ASP. NET+SQL Server 解决方案的主要安全隐患来自操作系统、开发平台以及后台数据库之间的无缝连接问题上,其他还有网页设计过程中的安全意识、数据库自身的安全性等问题。在 Web 系统开发过程中常见的安全问题主要有以下几类:

(1)需求分析、框架设计阶段留下的安全隐患。目前很多软件系统开发者在软件需求分析时过于看重功能和质量、效率等问题,没有明确用户对安全性方面的需求,这将造成软件开发从一开始就缺乏安全性的考虑,可能会对软件开发完成后的使用和维护留下巨大安全隐患。

(2)程序设计过程中的安全隐患。主要有以下几种:代码中存在漏洞,如缓冲区溢出、对入口参数没有有效性检查等;开发人员在程序中植入恶意代码或故意留下"后门";引用别人的代码或公共源码,但没有经过安全性测试;开发人员使用的库函数存在安全漏洞等。

(3)管理方面的安全隐患。如代码存放不安全、开发文档不安全、开发的网络环境不安全,或是开发人员在不安全的地方修改、维护、讨论代码,不慎或故意泄漏开发项目信息、软件加密方法等,或者是未经许可,采用没有通过合法渠道获得的代码等,这些都可能对软件系统的安全性带来负面影响。

具体到基于 ASP. NET 开发的 Web 系统来说,一是源代码的安全性隐患。由于 ASP. NET 程序采用非编译性语言,大大降低了程序源代码的安全性。如果黑客侵入站点,就可以获得 ASP. NET 源代码;同时对于租用服务器的用户,因个别服务器出租商的职业道德问题,也会造成 ASP. NET 应用程序源代码的泄露。二是程序设计中容易被忽视的安全性问题。ASP. NET 代码使用表单实现交互,而相应的内容会反映在浏览器的地址栏中,如果不采用适当的安全措施,只要记下这些内容,就可以绕过验证直接进入某一页面。因此,在验证或注册页面中,必须采取特殊措施来避免此类问题的产生。

2 基于 ASP. NET 技术的 Web 应用系统的安全体系

目前在软件开发实践中更多的侧重对软件质量、功能等方面的控制,如 CMM 评级等措施,却对软件开发的安全问题较少问津。事实上,如果要保证软件系统的安全可靠,进行必要的软件安全控制也是非常重要的,尤其是对于面对成千上万不确定用户的 Web 系统而言。软件的安全控制应该有机融入软件系统开发的全过程,即首先了解软件开发中可能出现的安全问题,明确开发中的安全需求;然后保证程序设计、编码的安全性,并对开发出的软件进行安全性检测和评估。在整个开发过程中,还应当对开发人员进行安全培训和开发过程的全程监理。

一个安全的 Web 应用系统在开发过程中应该充分重视

赵 争 硕士,主要研究方向:软件工程、现代物流、企业信息化、企业建模、基于 Web 的软件开发。

操作系统、开发平台和后台数据库之间的安全衔接上，充分利用三者各自的安全机制，并通过联合机制形成强大的安全合力，以此确保整个 Web 应用系统的安全使用和运行。操作系统的安全机制自不必言，只要保证系统安装了有效的实施监控软件并处于系统防火墙的有效防护之下即可，SQL Server 数据库所带有的自身安全保护机制也使其具有较高的安全性，下面重点探讨开发平台 ASP. NET 的安全机制及其与操作系统之间的安全协作机制。

2.1 操作系统与开发平台的安全协作机制

在基于 ASP. NET 技术的 Web 应用系统中，IIS 就是 Windows 操作系统的代表，安全机制的实现及用户鉴别的处理是与 IIS 相互关联的。ASP. NET 鉴别是对来访用户或实体的身份进行权限的校验和确认。理解 ASP. NET 与 IIS 之间的相互关系有助于开发者设计出合适的安全系统。IIS 和 ASP. NET 以及操作系统提供的底层安全服务一起，提供身份验证和授权机制。IIS 是在 metabase 中保存其安全参数设置，而 ASP. NET 的安全性设置保存在 XML 配置文件 Web. config 中。安全的设计模式需要 IIS 与 ASP. NET 两者都能正确配置，而这两者是相互独立的，IIS 通过它自己的机制来确保一个用户访问系统资源时被鉴别。二者协作提供安全服务的基本原理是：当服务器接收到来自客户的 HTTP(S)请求后，IIS 使用基本、摘要、集成或证书对调用方进行身份验证。若将 IIS 配置为使用匿名身份验证，则网站上的内容不需要经过身份验证即可访问。通过验证以后，IIS 用户创建一个访问令牌。这一步的授权通过 NTFS、特定 IP 地址等方式进行。

2.2 开发平台的安全机制

IIS 将已验证的调用方访问令牌传递到 ASP. NET，由 ASP. NET 对调用方进行身份验证。ASP. NET 可以使用 Windows 身份验证、表单身份验证、Passport 身份验证等方式对用户进行身份验证。Windows 身份验证依靠 IIS 对用户进行身份验证并创建 Windows 访问令牌来表示经过身份验证的标识。如果不需要对客户端进行身份验证(或者实现自定义的身份验证方案)，Web 服务器将创建 Windows 访问令牌来表示使用同一个匿名帐户的所有匿名用户。授权访问则通过两个模块进行：UrlAuthorizationModule 模块使用 Web. config 中配置的授权规则确保调用方访问所请求的资源，FileAutho-rizationModule 模块在 Windows 身份验证中检查调用方是否具有访问所请求资源的权限。

3 数据库资源的安全访问机制

Web 应用系统中的重要资源如用户数据库、文档资料数据库、流媒体资源等不允许普通用户直接操作和下载，这就需要对站点的资源进行保护。在 Web 应用系统中，最容易出现安全问题的环节就是对数据库的访问，包括读处理和写处理，其中如何处理与数据库的连接及连接字符串如何保存等问题是一个关键。在 Oracle 及 SQLserver2000 这样的数据库中有两种处理方式，或者通过 Windows 操作系统进行处理，或者通过数据库提供安全机制进行处理。前者安全性较低，而使用 Windows 认证去访问数据库的方式提供了较高的安全性，这种方式下使用连接串但不需要在连接串中存放密码。大多数基于 ASP. NET 的 Web 应用在 Web. config 或 global. aspx 文件中存储连接串。从某种程度上来讲，Web. config 是安全的，因为它不允许被客户端浏览或下载。然而，这些文件以明文的方式存在限制了它的安全性，因此它不是存储连接串信息的最好场所，入侵者很容易访问这些文件并危及到 Web 服务器的安全。为确保对受保护资源的安全访问，本文建立了一个数据库资源安全访问机制，利用用户提供的网上登录身份证明，如 E-mail 和口令，通过相应的身份鉴别程序进行登录操作。如果系统需要对请求进行鉴别，ASP. NET 可以发送一个包含身份证明的 cookie。接下来的请求都在包头含有这个 cookie，以此来证明身份验证已经通过，无须再次验证。需要注意的是形式本身并不能起到身份验证的作用，只是被提供作为一种获得用户身份的方法，对 E-mail 和口令的处理过程才能够实现对用户身份的鉴别。整个用户身份鉴别过程的运行机制如图 1 所示。

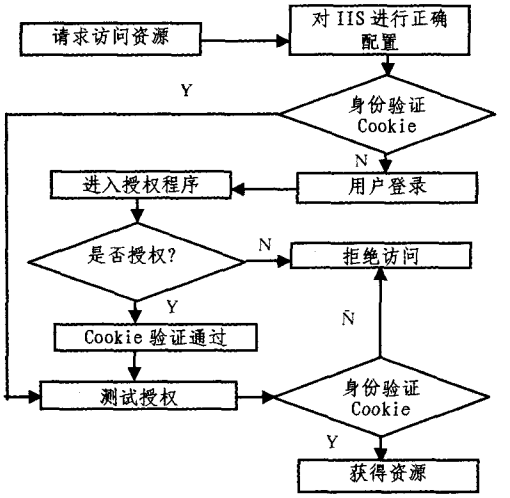

图 1 Web 应用系统中的数据库资源安全访问机制

4 带有资源保护机制的文档管理系统

4.1 系统概述

在信息技术迅猛发展的今天，面对大量的信息资源，怎样提高对各种电子文档的利用率已成为企业和个人迫切需要解决的问题。本系统的设计目的就是构建一个“电子书架”，以实现高效、安全、集中地管理单位内部的各种文档资料，还可以通过网络化的管理共享丰富的信息资源。目前市面上也有各种商用文档管理系统，但价格相对昂贵。本系统通过简化功能和人性化的界面更加适用于中小企业的文档管理。特别值得一提的是，本系统采用了安全的数据库访问机制，可以有效提高数据库资源的安全性。

系统可分为以下几大模块：

(1)用户登录模块。判断用户是合法用户还是超级管理员，然后跳转到相应的页面，根据用户类别加载用户管理模块。用户分为两类：普通用户与超级管理员。普通用户登录系统之后可以浏览查询文档，发表文档，并对文档进行编辑、删除。超级管理员登录系统后，可以浏览所有文档(包括隐藏的文档)，并对文档进行删除操作。除此之外，超级管理员还可以修改自己的个人信息，并对所有的用户进行管理。

(2)文档管理模块。这是文档管理系统的核心模块。用户登录之后可以浏览自己所发的文档，并对其进行编辑、删除等操作，还可以浏览所有可见的文档，但不一定对所有文档都有删除权限，另外，还可以发表新文档，并按一定的条件搜索相应的文档。

(3)用户注册、重登录和退出模块。访问者可通过在此模块输入个人信息成为注册用户，或者重新登录以及退出文档管理系统即清除用户的会话状态。

(4)用户管理模块。此模块是超级管理员特有的，如果是

超级管理员登录后会自动加载该模块。包括对访问者提交的行注册资料进行审核，将符合申请条件的访问者审批为普通用户。

4.2 文档数据库的资源保护

文档管理系统的开发主要包括后台数据库的建立和维护以及前台应用程序的开发两部分。系统采用 ASP. NET 作为前台开发工具，Microsoft SQL Server2000 作为后台数据库开发平台。通常我们采用 ADO. NET 来完成前台程序与后台数据库的交互。ADO. NET 是一组用于和数据源进行交互的面向对象类库。通常情况下，数据源是数据库，但也可以是 TXT 文件、Excel 表格或者 XML 文件。本文采用的 Microsoft SQL Server 2000 主要有以下几种主要对象：SqlConnection，SqlConnection，SqlDataReader，SqlDataAdapter 和 DataSet 对象。为防止数据库被下载，在设计与数据库的连接时，可使用 ODBC 数据源。在 ASP. NET 程序设计中，如果有条件，应尽量使用 ODBC 数据源，不要把数据库名写在程序中，否则，数据库名将随源代码的失密而一同失密。如果使用 ODBC 数据源，就不会存在这样的问题了。为有效地防止源代码泄露，可以对 ASP. NET 页面进行加密。可采用两种方法对页面进行加密。一是使用组件技术将编程逻辑封装入 DLL 之中；二是使用微软的 Script Encoder 对页面进行加密。使用组件技术存在的主要问题是每段代码均需组件化，操作比较繁琐，工作量较大，而使用 Encoder 对页面进行加密，操作简单、收效良好。由于本文重点讨论系统的安全问题，因此只给出实现数据库安全访问过程的基本思想。

用户在没有经过身份验证前，如果试图访问各保护子目录中的文件，如在浏览器的地址栏中写入受保护文件名等，都被重定向到子目录下的登录网页 loginchild. aspx。只有从该登录网页进入，才可以打开子目录链接页 indexchild. aspx，并由此页打开其他的受保护页面。要实现这种安全机制，首先需要在站点根目录下的 Web. config 配置文件中进行如下设置：

```
〈authentication mode="forms"〉
〈forms LoginUrl= "index. aspx" name= "login" timeout="60"
  path="/"〉〈/forms〉
〈/authentication〉
```

其次，为了保护子目录中的文件，可通过根目录中 Web. config 配置文件的〈location〉标记对子目录指定不同的安全设置，对相应用户进行授权，从而使匿名用户不能访问，限于篇幅，源代码从略。

结束语 本文提出并设计了一个基于 ASP. NET 的文件管理系统原型，重点对有效保护系统的数据库资源进行了分析和设计。该系统整合了数据库技术、ASP. NET 技术及网页文档编辑技术等，实现了对组织内部体系文件的全面管理，提供了不同层次的需求服务，从而使文件管理更加富有效率，维护成本大幅降低。基于 ASP. NET 的 Web 应用程序开发，在充分利用 ASP. NET 的安全机制、数据库安全控制、管理员网络安全防范意识的基础上，极大地提高了文档管理系统的安全性能。但本系统还存在一些不足之处，如普通记录文件的更新与完善需要加入、查询的智能性需进一步提高等，这是该系统需要进一步修改和完善的内容。

参 考 文 献

1 邓良松. 软件工程. 西安：西安电子科技大学出版社，2000
2 吴晨. ASP. NET 数据库项目案例导航. 北京：清华大学出版社，2004
3 林菲. 软件工程中的需求分析. 福建电脑，2005(7)：80～81
4 唐艳，段洪君. 用 Internet 软件工程方法开发软件. 牡丹江教育学院学报，2005(3)：99

(上接第 59 页)

(1)标识和托肯。系统标识是 Petri 网格系统中所有网格资源标识的集合。系统中的托肯是网格资源 P 中的所有信息分量，系统中的所有托肯集合为 $\bigcup_{p\in P} Type(p)$。系统中的网格资源 P 的资源标识就是该资源中的托肯集合 $M(p)$，它是一个多重集合。而系统托肯则是系统中全部资源的标识的集合 $\sum_{p\in P} M(p)$，是系统中全部资源多重集的多重集。

(2)服务前提。一个服务的发生完全由它的外延(服务前集与服务后集之和)决定，而与系统全局状态无关。服务前集是指由输入弧与服务连接的资源集合，服务后集是指由输出弧与服务连接的资源集合。服务发生又可分为单个服务模式和并发服务模式两种情形。

(3)单服务模式。对于当前服务的所有服务前集，都有 $Pre(se)\leqslant M$。服务前提满足时，服务就自动发生。服务发生时，前集中的托肯相应减少 $Pre(se)$，后集中的托肯相应增加 $Post(se)$，即单个服务发生后的系统标识为：$M' = M - Pre(se) + Post(se)$。

(4)并发服务模式。对于并发服务的所有服务前集，都有 $Pre(Se_\mu)\leqslant M$。服务前提满足时，服务就自动发生。服务发生时，所有服务前集中的托肯相应减少 $Pre(Se_\mu)$，所有服务后集中的托肯相应增加 $Post(Se_\mu)$，即并发服务发生后的系统标识为：$M' = M - Pre(Se_\mu) + Post(Se_\mu)$。

$Pre(Se_\mu) = \sum_{se\in SERVICES} se\mu(se)Pre(se)$。所有在 Se_μ 中的 Web Services 都被认为是并发的网络服务，当 $|Se_\mu| = 1$ 时，并发服务模式退化为单个服务模式。

(5)服务规则。如果一个并发服务模式的有限多重集 Se_μ，在标识 M 下是激活的，则一步执行可以产生新的标识 M'，$M' = M - Pre(Se_\mu) + Post(Se_\mu)$。表示为 $M[Se_\mu > M'$ 或 $M \xrightarrow{Se_\mu} M'$。因此，一步执行既可以包括单个服务模式，也可以包括多个并发服务模式。

结论 目前，网格系统还没有一种准确、清楚和形式化的定义。本文探索性地给出了网格系统的一个高层抽象模型，即有色网系统。利用 Petri 网工具来构架网格系统模型是网格系统理论深入研究和应用的必然，因为网格正是 Petri 网格系统依赖的完全异步并发的基础环境，基于 Petri 网原理建造的 Petri 网格系统能更好地模拟现实社会，建立可伸缩的虚拟组织。在这个定义之上，有关网格系统的性质和行为的描述是我们下一步的研究工作。

参 考 文 献

1 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations[J]. International J. Supercomputer Application, 2001, 15(3)
2 Foster I, Kesselman C 著. 网格计算(第二版)[M]. 金海，袁平鹏，等译. 北京：电子工业出版社，2004
3 徐志伟，冯百明，李伟. 网格计算技术[M]. 北京：电子工业出版社，2004
4 袁崇义. Petri 网原理与应用[M]. 北京：电子工业出版社，2005
5 Web Services 工作组. Web Services Activity [EB/OL]. http://www. w3. org/2002/ws/
6 Czajkowski K. The WS-Resource Framework[EB/OL], http://www. gridhome. com/dvnews/show. aspx? id=70&cid=40
7 赵维. TaoGrid 服务资源网格[EB/OL]. http://www. chinagrid. net/dvnews/upload/2004_11/04111615512121. doc