Doi:10.3969/j.issn.1003-5060.2013.04.013

一种基于 B/S 架构的安全数据下载方法研究

熊宗武[1]，　石　雷[2]，　周国祥[2]

（1.铜陵市住房公积金管理中心，安徽 铜陵　244000；2.合肥工业大学 计算机与信息学院，安徽 合肥　230009）

摘　要：文章探讨了一种基于 B/S 架构系统的安全数据下载方法，并比较了几种可行方案的优缺点，从而说明了文中提出的“下载即删除”的安全数据处理方法可从根本上保证文件下载的安全性。但是，由于浏览器自身功能的限制使得该方法无法直接实现。为此，文章设计了一种流文件传输的变通方案，更好地实现了文中所提出的安全数据处理方法。目前，该方法已在某市住房公积金管理系统中得到应用，并取得了良好效果。

关键词：管理信息系统；B/S 架构；安全数据下载；流文件传输

中图分类号：TP317.1　　**文献标志码**：A　　**文章编号**：1003-5060(2013)04-0444-04

A secure download method based on B/S structure

XIONG Zong-wu[1]，　SHI Lei[2]，　ZHOU Guo-xiang[2]

(1. Tongling Housing Fund Management Center, Tongling 244000, China; 2. School of Computer and Information, Hefei University of Technology, Hefei 230009, China)

Abstract: A secure download method is given in this paper which can ensure the security of the system based on B/S structure. The advantages and disadvantages of several possible options are compared, and it is shown that the method of download and delete proposed in the paper can fundamentally ensure the security of data processing. However, due to the restrictions on browser, the method can not be used directly. Then an alternative method based on the streaming file transfer program is proposed to realize the secure data processing method proposed. This method has been applied effectively in a city's housing fund management system.

Key words: management information system; B/S structure; security data download; streaming file transfer

0　引　　言

随着网络技术、数据库技术的发展，现代管理信息系统已经由传统的 C/S 模式发展为 B/S 模式[1]。在 B/S 架构下，用户工作界面通过浏览器来实现，主要事物逻辑则在服务器端实现，通过服务器端直接访问和操作数据库[2]。因此 B/S 架构下的管理信息系统最大的优点是可以在任何地方进行操作而不用安装任何软件，只要有一台能上网的电脑即可。

然而，由于超文本传输协议（HyperText ransfer Protocol，简称 HTTP）和浏览器的天然特性决定了 B/S 架构下的系统中信息泄露和被非法攻击的可能性很大，因此在进行 B/S 架构系统设计时必须考虑系统的安全性问题[3]。

目前，大部分 B/S 系统在设计安全模块时均采用了基于角色的访问控制（Role Based Access Control，简称 RBAC）方法[4-5]，即为不同的用户赋予不同的权限，不同的权限能够访问的数据和操作不同。RBAC 模型如图 1 所示。采用这种方

收稿日期：2012-11-07；修回日期：2013-02-27
基金项目：合肥工业大学博士学位人员专项基金资助项目（2012HGBZ0633）
作者简介：熊宗武（1967－），男，安徽歙县人，铜陵市住房公积金管理中心工程师；
周国祥（1956－），男，安徽合肥人，合肥工业大学教授，硕士生导师.

式可以很好地满足 B/S 系统中的一般安全性要求，但是当遇到黑客攻击、木马盗取等事件时，该模型将无效。

另一种常用的方式是对用户要访问的关键数据进行动态加密[5]。当远程客户端提出数据访问请求时，服务器端首先根据远程客户端的信息（如登录账户名、登录 IP 地址等）生成一动态密钥，并用此密钥对关键数据加密后再发送到客户端，客户端只有通过自己的密钥才能将数据解密出来。其他客户端如果获取了该加密数据，由于不知道动态密钥无法对其解密，从而保证了系统的安全性。其工作原理如图 2 所示。

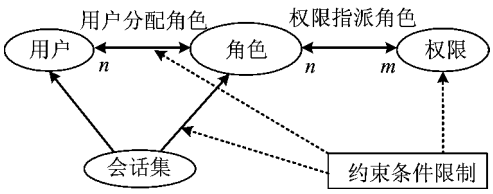

图 1　RBAC 模型

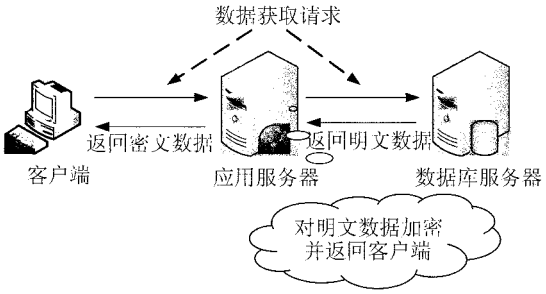

图 2　对数据动态加密流程

采用动态加密的方式可以解决黑客攻击等问题，但也有其不足之处。应用服务器每次都必须对要返回给客户端的数据动态加密，当要返回的数据量较大时，或者并发的用户数较多时，应用服务器的工作量将会很大，从而造成整个系统性能的下降。因此如果客户端请求的是较大的文件信息时，这种方式也不适合[6]。而在很多办公自动化（office automation，简称 OA）系统中，请求的数据又往往是一些文件信息。

本文结合某市住房公积金单位用户信息上报系统，探讨 B/S 架构下一种基于 HTTP 协议的安全数据下载方法。

1　系统建设需求

某市住房公积金管理中心经过几年的信息化建设，已经构成了 2 个核心数据库、3 个服务逻辑层、4 个应用支撑组件的完整管理平台。其中，2 个核心数据库指业务信息数据库和基础信息数据库，存放支撑整个平台的各种业务信息和全市职工公积金的各种数据信息；3 个服务逻辑层指应用浏览层、Web 应用服务层和数据服务层，分别对应前台的页面显示，中间的业务逻辑处理和底层的数据库服务；4 个应用支撑组件指业务应用逻辑、系统管理应用逻辑、组合查询应用逻辑和基金信息的交换与共享平台。该平台采用 C#.Net＋Sybase/SQL Server 开发，整个平台的架构如图 3 所示。

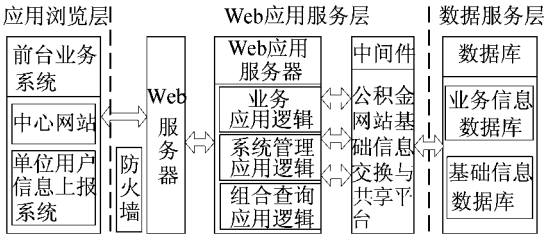

图 3　某市公积金管理中心信息化管理平台逻辑架构图

其中，单位用户信息上报系统是提供给各单位会计使用的上报职工公积金信息和业务报表的电子化系统。在每年的 3—4 月份，各单位需集中将本年度本单位上缴公积金的职工工资、上缴额度等各种信息通过该系统上报，上报内容集中体现为 3 张表格，即《缴存额调整清册》、《缴存额调整申报表》和《缴存额调整审批表》。这 3 张报表需要先由单位用户填报，填写完毕后再下载由系统自动生成的 Word（或 Excel）文档打印盖章后送交公积金管理中心，最后由公积金管理中心审核并备案。《缴存额调整申报表》的填报界面如图 4 所示。

图 4　《缴存额调整申报表》的填报界面

2　安全下载方法探究

“单位用户上报系统”的核心是将信息填报后

存入数据库,并根据数据库中的存入数据生成Word(或Excel)文档供用户下载。生成Word(或Excel)文档可借助于微软的Office程序集Microsoft.Office.Interop.Word/Excel来完成,生成的文档必然会保存在服务器的某个文件夹内,因此目前的问题是如何访问该文档,以及如何保证该文档在当前的访问模式下不会被非法下载。本文讨论的几种可行方法及其优缺点如下。

(1)随机文件名方式。随机生成文档的文件名,并提供用户静态链接地址供下载。

例如可根据文档产生的时间戳对生成的文档命名,并提供用户下载的静态链接地址。采用此种方法简单,代码上易实现,同时具有一定的安全性保证。由于文件名是随机生成的,非法访问者不能轻易猜测到,因此不会轻易破解并下载。

然而,由于采用静态链接地址的形式,非法访问者可以轻易得到文件存放在服务器上的具体目录位置,从而可以通过暴力搜索的方法非法下载文件。尽管可以通过一些安全性设置将远端目录设为不能直接访问,但不能从根本上解决该问题,因此该方法存在较大的隐患。

(2)隐藏真实文件名方式。远端下载时隐藏文档的真实目录和文件名,下载完毕后用指定的名称重命名远端文档的副本。

由于隐藏了文档的真实目录和文件名,非法访问者无法知道文件存放在服务器上的具体目录位置,因此采用这种方式可有效解决随机文件名方式的不足。采用隐藏真实文件名方式的实现代码略复杂,动态生成文件名必须在服务器端完成。一般地,可采用.NET程序集中的HttpResponse类来实现。

然而,尽管隐藏了文档的真实目录和文件名,但非法访问者仍然能通过一些技术找到真实目录从而非法下载文件,因此该方法仍然存在着安全隐患。

(3)下载完毕即删除方式。当客户端提出文档下载要求时生成文档,下载完毕后删除文档。

在该系统中,需要生成的3张报表的数据都存放在数据库中,用户下载的文档是根据数据库中的内容实时生成的,因此下载完毕后,文档没有必要继续保存在服务器上,完全可将其删除。下次客户端再有文档下载请求时可再根据数据库中的内容实时生成新的报表。由于文档的生存期很短,采用该方式可有效防止非法用户的非法下载,从而彻底解决问题。该方式的程序流程如图5所示。

然而,虽然"下载完毕即删除"方式可彻底解决存在安全问题,但由于服务端HTTP协议的限制使得该方法几乎无法实现。本文将提出一种变通的方式巧妙解决该问题。

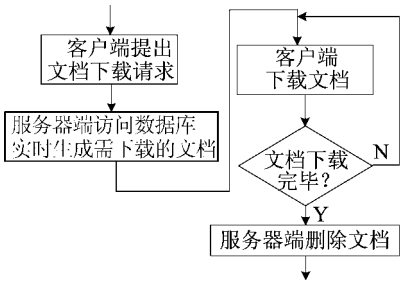

图5 下载完毕即删除方式的流程图

3 "下载完毕即删除"的HTTP实现

HTTP采用面向连接的TCP/IP协议实现客户端和服务器之间的会话过程,因此一个典型的HTTP文件下载过程为:

(1)客户端向服务器端建立Socket连接,指定服务器端的IP地址和端口号。

(2)客户端向服务器端发出GET请求包请求下载文件,包括文件名称和文件所在目录信息等。

(3)服务器端对客户端作出响应,返回请求响应包,包括文件大小、文件类型等。

(4)服务器端采用TCP协议传输文件。

(5)文件下载完毕后双方结束会话,关闭连接,释放占用的资源。

在ASP.NET框架下客户端和服务器端交互有2个重要的类,分别是HttpResponse和HttpRequest。其中HttpResponse类是ASP.NET框架下服务器端提供给客户端当前页面输出流的访问,而HttpRequest则是用以将客户端请求的信息提交给服务器。亦即一个输入,一个输出。HttpRequest类通常是在客户端浏览器执行了GET/POST方法后在页面刷新时服务器端的执行代码中调用。即如果采用HttpResponse/HttpRequest方式实现文档的下载删除,应该在以上代码执行完毕后由客户端浏览器发送GET/POST命令,强制页面刷新,并在页面刷新代码中检测HttpRequest的返回值,如果确定客户端下载文件完毕,则由服务器端删除生成的临时文件。

然而这种实现存在2个问题[7]:

（1）客户端浏览器无法检测文档下载完毕。由于客户端浏览器只能执行 Javascript 脚本语言，实现一些页面美化和简单的交互操作，无法检测浏览器下载文件状态这种复杂的工作。

（2）强制页面刷新是一种不友好的操作方式。由于页面刷新会使得用户体验变差，现在主流的 AJAX 等技术均尽可能避免页面刷新，因此采用 GET/POST 方式强制页面刷新会带来页面不友好的操作方式。

因此以上思路不可行，则必须构造一种方法，使得不需要客户端给出下载状态信息服务器端就可以判断客户端下载完毕与否，并作出正确的处理。

在采用 HTTP 协议进行数据传输时，数据是一份一份从服务器端发送到客户端的，只不过由于 ASP. NET 良好的封装，在上层的开发者感受不到这种发送方式。而如果上层直接将大文件打散成若干小的数据包并一份一份发送到客户端，服务器端就可以根据已发送出的数据包的份数间接判断文件是否发送完毕。而这种打散发送的方式正是流文件的操作方式[8]，在 ASP. NET 中有专门的函数支持。采用这种思想，“下载完毕即删除”的方式流程图重构如图 6 所示。

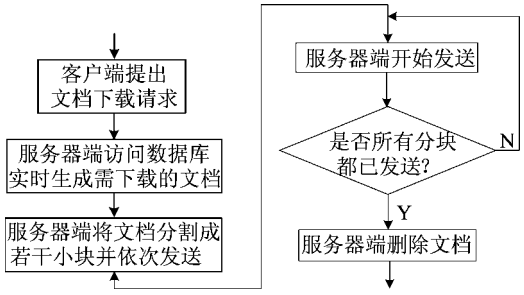

图 6　下载完毕即删除方式的服务器端实现流程图

实现关键代码如下：

```
_BinaryReader. BaseStream. Seek ( startBytes, SeekOrigin. Begin);
int maxCount = (int) Math. Ceiling((DownloadFile. Length - startBytes + 0. 0)/1024);
for (int i = 0; i < maxCount && System. Web. HttpContext. Current. Response. IsClientConnected; i++)
{
System. Web. HttpContext. Current. Response. BinaryWrite (_ BinaryReader. ReadBytes (1024)); System. Web. HttpContext. Current. Response. Flush();
}
```

以上代码执行完毕后，再采用服务器端的文件操作函数将临时生成的文档删除即可。

4　结束语

在 B/S 架构的系统中，客户端可能要求服务器端生成一些特定的文档供下载使用，而这些生成的文档如果不及时回收处理就会造成一些安全上的隐患。本文探讨了一种“下载完毕即删除”的方式可以有效解决该安全隐患。这种方式实现的最大难点是服务器无法判断客户端是否已经将文件下载完毕从而进行删除操作。为此，本文又提出了一种将文件分割发送的流式传输方式有效解决了该难点。本文所建立的解决方案已经在某市公积金业务系统中使用，并取得了很好的效果。

[参　考　文　献]

[1] Kou C Y, Springsteel F. The security mechanism in the World Wide Web(WWW) and the common gateway interface(CGI)[C]//Proceedings of IEEE 31st Annual 1997 International Carnahan Conference on Security Technology, 1997:114－119.

[2] Chen Y M, Zhao J, Zou Y B, et al. Office automation system for enterprise based-on . NET[J]. Journal of Donghua University: English Edition, 2010, 27(4): 522－529.

[3] 刘孝保，杜平安. B/S 环境下 CIMS 安全模型设计与实现[J]. 电子科技大学学报，2008，37(1)：109－112.

[4] Sandhu R, Ferraiolo D, Kuhn R. The NIST model for role-based access control: towards a unified standard[C]//Proceedings of 5th ACM Workshop on Role-based Access Control, Berlin, Germany, 2000: 47－63.

[5] 张仁斌，朱　飞，吴燎原. 一种嵌入式 Linux 系统的身份认证方法[J]. 合肥工业大学学报：自然科学版，2009，32(8)：1158－1161.

[6] 肖　达，舒继武，薛　巍，等. 基于组密钥服务器的加密文件系统的设计和实现[J]. 计算机学报，2008，31(4)：600－610.

[7] Lee D, Kim K, Yoon T B. Design of web page evaluation system using Ajax and neural networks[C]//Evolutionary Computation, Hong Kong, China, 2008: 3025－3029.

[8] 吴峰光，奚宏生，徐陈锋. 一种支持并发访问流的文件预取算法[J]. 软件学报，2010，21(8)：1820－1833.

（责任编辑　闫杏丽）