基于J2EE架构的安全电子商务/电子政务系统

孙秀红[1],易泽湘[1],易锦华[2]
(1.武汉理工大学 计算机学院,湖北 武汉 430070;
2.华夏银行北京总行 个人金融部,北京 100032)

摘 要:介绍了Java平台安全、J2EE安全体系结构,用一个完整的电子商务请求流程为例研究了J2EE核心组件技术(JSP/Servlet,EJB)如何在安全电子商务/电子政务系统中发挥作用,以及如何用J2EE安全体系结构架构安全电子商务/电子政务系统。

关键词:Java;J2EE;JSP/Servlet;电子商务/电子政务
中图分类号:TP393.08 **文献标识码**:A **文章编号**:1673-629X(2006)09-0204-04

Security E-Commerce/E-Government Based on J2EE Framework

SUN Xiu-hong[1], YI Ze-xiang[1], YI Jin-hua[2]
(1. School of Computer Science & Technology, Wuhan University of Technology, Wuhan 430070, China;
2. Individual Finance Dept. of Huaxia Bank, Beijing 100032, China)

Abstract: Discuss the platform security of Java, the security architecture of J2EE. Research the key module technology of J2EE(JSP/Servlet, EJB) how to display use in the system of security E-Commerce/E-Government and how to build the security E-Commerce/E-Government system by the security architecture of J2EE through an example with E-Commerce requiring flow.

Key words: Java; J2EE; JSP/Servlet; E-Commerce/E-Government

0 引 言

Internet和Intranet的迅速发展有力地推动了商业和政务的电子化,电子商务和电子政务的发展又反过来进一步推动了因特网的迅速发展,企业和政府为了扩展业务范围、降低经营成本、缩短和客户之间的响应时间,迫切需要搭建一个安全可靠的商务/政务平台。J2EE一个开放的、基于标准的开发和部署的平台,为企业级应用的设计、开发、集成以及部署提供了一条基于组件的实现途径,并提供了高性能、高可靠性和可伸缩性的运行支撑环境。

1 Java平台的安全

J2EE安全体系结构建立在J2SE的基本特征之上,而J2SE又是以Java平台为基础,在研究J2EE架构安全电子商务/电子政务系统之前,有必要了解Java平台安全机制和J2EE的安全体系结构。Java语言是一种与操作系统平台无关的语言,本身有着较为完善的安全机制,用Java开发的应用程序,可以安全地在Internet上运行。最初的Java平台(JDK1.1)采用沙箱(sandbox)安全模型,基本安全模型的核心主要由3个支柱承担,这3个支柱是Java运行环境的3个重要安全组件,分别是:类加载器、类文件验证器、安全管理器。下面以简化的JVM(Java Virtual Machine)图(见图1)来分析Java平台如何实现自身安全[1]。

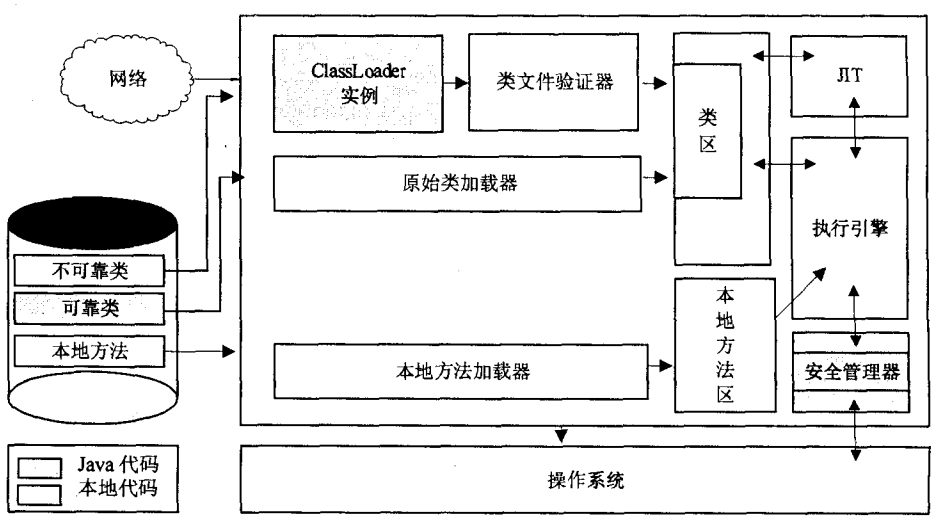

图1 JVM组件安全模型

JVM组件安全模型中的类加载器决定Java程序如何

收稿日期:2005-12-29
作者简介:孙秀红(1980-),女,山东德州人,硕士研究生,研究方向为网络数据库。

以及何时加载代码,并且最终负责加载过程。从安全观点来看,类加载器确保运行环境中的系统组件不会被不可靠的代码所替代。而类文件验证器则负责校验字节码是否违反JVM的类型安全限制(符合安全限制则内存不会发生上溢或下溢,字节码指令也将拥有正确的类型参数)来确保非系统代码正确格式化[2]。而安全管理器作为java.lang.SecurityManager一个实例来执行,主要完成如下工作:当企图执行文件和网络I/O操作时加强运行访问控制限制,创建新的类加载器,操作java.lang.Threads和java.lang.ThreadGroups,启动操作系统上的进程,终止JVM,把非Java库——本地代码——加载到JVM中,执行某些类型的窗口系统操作,把某些类加载到JVM,实例化安全管理器,改变当前的安全管理器,访问系统和安全属性。但是沙箱安全模型有其自身的局限性,如安全策略和实施分离的不彻底性、内部安全机制的脆弱性等。于是引入一些新的安全机制,加强Java平台的安全性。新的安全模型中主要部件包括安全策略、访问许可权、保护域、访问控制核查、优先级操作和Java类的加载和实施方案。当前Java平台(J2SDK)支持灵活的实施访问控制的安全策略,同时具有可扩展性和规模可变性,可以使用安全策略来决定赋予它的运行代码何种访问权。Java平台作为基础的开发平台,其提供的加密组件接口与安全服务解决方案,开发复杂的企业级分布式应用程序功能十分强大,J2EE规范及其相应的安全机制又进一步扩展了Java平台安全方面的功能,因此Java成为开发Web应用与企业电子商务和政府电子政务解决方案的首选语言。

2　J2EE的安全体系结构

Java较为完善的安全机制,还远远不能满足企业/政府对系统的至高安全要求,正因如此,J2EE规范及其相应的安全机制进一步扩展了Java平台安全方面的功能。在企业环境中,J2EE提供了一个开发标准,应用程序无需对安全策略进行硬编码,使得组件开发者并非安全专家,而能开发出安全性很高的商务/政务系统应用程序,此种应用程序可以与声明性的安全策略捆绑在一起,放进应用程序组件的装配集。使用J2EE安全模型的安全策略可以应用于兼容J2EE的任何操作环境,并且,可以部署在任何兼容J2EE的应用程序服务器中。如下具有缓存反向代理服务器和安全插件的集群环境图[3](见图2)就充分体现了J2EE安全体系结构,为电子政务系统/电子商务系统增加了多层保护屏障:

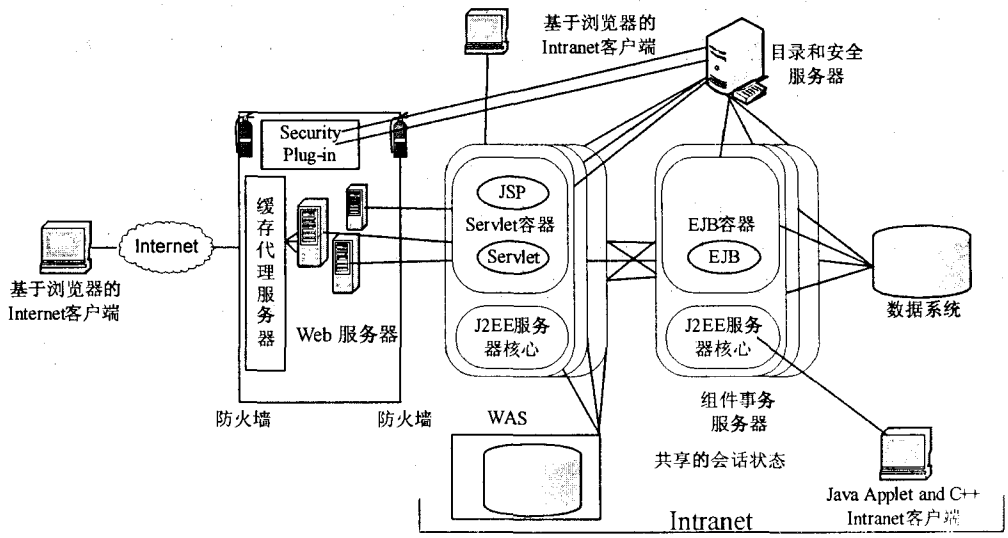

图2　具有缓存反向代理服务器和安全插件的集群环境图

①客户端在Internet/Intranet上流动的数据采取了(JSSL)加密技术;

②Web服务器架构双层防火墙;

③企业内部网络采取安全缓存代理服务器;

④Servlet/JSP页面显示层、EJB业务逻辑层、EIS层分别采取了鉴别、授权、数据完整性和保密性等机制。

撇开①②③,J2EE安全模型综合采用了鉴别、授权、数据完整性和保密性等机制,分别在Web层、EJB层和EIS(企业信息系统层)实现安全机制,保证了整个系统的安全。J2EE安全模型的最主要需求是支持安全的应用程序的部署,在应用程序运行期间,无需依赖私有的网络或其它独立的技术,使之在容器间具有可移植性,同时也减轻了应用程序开发者的负担,将安全责任移交给更专业的J2EE相关角色。由于篇幅有限,不讨论J2EE的鉴别机制、认证与授权机制、消息保护、审核、容器安全等的具体实现,而以一个具体的安全电子商务请求流程为例来研究J2EE如何通过其核心组件技术(JSP/Servlet、EJB的安全技术)来架构安全电子商务系统(电子政务系统)[3]。

3　J2EE架构安全电子商务系统实例

网上租赁电子商务系统,是用Windows 2000 Server+JBuilder2005+BEA Weblogic +Oracle9i搭建的基于J2EE架构的标准安全电子商务平台,用Weblogic作为应用服务器,来实现整个系统的J2EE架构,完成所有的业务逻辑和负载均衡。JSP/Servlet页面程序已经装载到Web容器,实体Beans(Account Bean)已经部署到EJB容器,以电子商务请求流程,来研究J2EE安全架构技术是如何在网上租赁电子商务系统中发挥作用的,如何利用J2EE安全架构技术来搭建安全电子商务/电子政务平台[4,5]。

3.1　电子商务安全

电子商务的一个重要技术特征是利用IT技术来传输

和处理商业信息。因此,电子商务的安全从整体上可以分为两类:计算机网络安全和商务交易安全。计算机网络安全包括:计算机网络设备安全、计算机网络系统安全、数据库安全等,如图 2 中的防火墙技术。商务交易安全则紧紧围绕传统商务在互联网上应用时产生的各种安全问题,在计算机网络安全的基础上,如何保障电子商务过程的顺利进行,即实现电子商务的保密性、完整性、可鉴别性、不可伪造性和不可抵赖性[5],如图 3 中的⑤,⑥采取的 JSSE 加密技术及 JAAS 授权和认证技术,网络安全技术和商务交易安全技术二者相辅相成,贯穿整个商务交易流程。

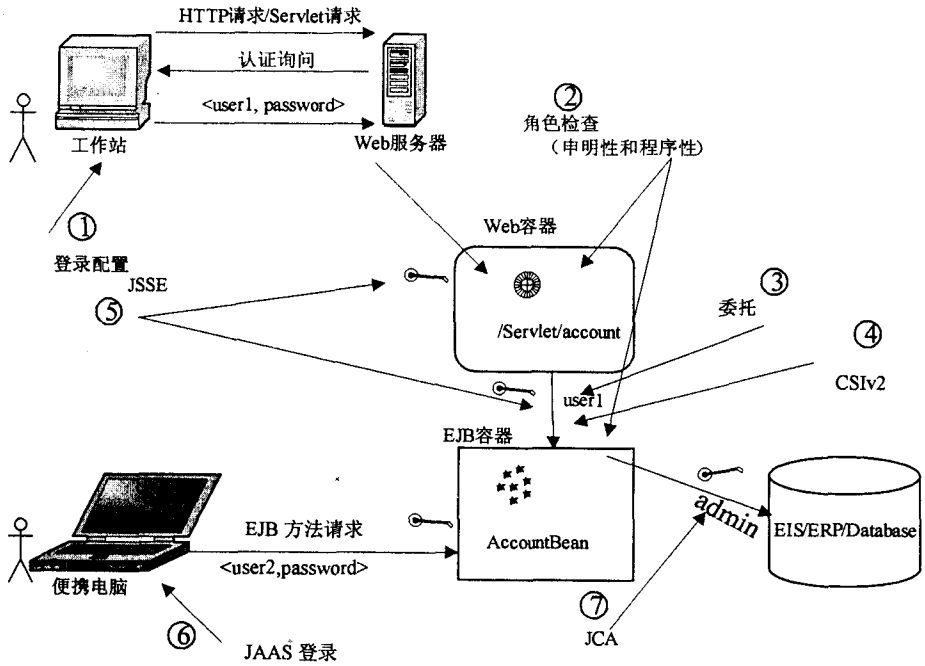

图 3 安全电子商务请求流程图

3.2 JSP/Servlet 安全

JSP/Servlet 技术像其他开发 Web 应用程序技术(ASP,PHP,CGI)一样存在着安全性因素,主要集中在由于服务器端特性和配置引起的安全问题,及 JSP 编程中引起的安全问题。当 JSP 页面第一次被调用的时候被编译成 JavaServlet 类,可以通过 JSP/Servlet 的认证、授权、主体委任、程序性安全 Web 组件的运行时约束来解决,如 user1(租赁者)通过登录取配置(步骤 1)调用 URL http://samples.com/servlet/account,当请求到达时 Web 容器使用底层的用户注册中心来认证用户 ID 和密码,当证书得到校验时,Web 容器就会执行授权检查。URI 以只授权 Teller 角色访问的方式受到保护。容器检查 user1 是否具有这个角色(参见图 3 的步骤②),因为 user1 具有 Teller 角色,所以他允许这个 URI。其登录配置和角色安全技术配置如下[6]:

基于表单的认证登录配置:

```
<login-config>
<auth-method>FORM</auth-method>
<form-login-config>
<form-login-page>/login.jsp</form-login-page>
<form-error-page>/login-failed.html</form-error-page.
</form-login-config>
</login-config>
```

基于证书的认证登录配置:

```
<login-config>
<auth-method>CLIENT-CERT</auth-method>
<login-config>
```

安全角色 Teller 和 Supervisor 的部署描述:

```
<assembly-descriptor>
<security-role>
<description>
This role is intended for employees who provide services to customers (tellers)
</description>
</security-role>
<security-role>
<description>
This role is intended for supervisors.
</description>
<role-name>Supervisor</role-name>
</security-role>
</assembly-descriptor>
```

Servlet 调用 AccountBean EJB,请求被分派到 EJB。Servlet 的委托策略并未被设置 run-as,这意味着下次请求可以使用调用者的身份执行。在这种情况下,调用者是 Bob,因此调用 EJB 所使用的身份由 user1 的证书组成,如图 3 中步骤③。

3.3 EJB 安全

(1)EJB 技术定义了一组进程,从应用程序的开发一直到被部署组件的管理,EJB 规范描述了一套从软件组件的开发到部署的约定,并勾画出了整个处理进程,EJB 的每个角色在组件的开发、部署和管理的不同阶段承担着自己的职责。这些职责包括:每个组件的事务特征和安全特征的定义和管理。其角色分别是:EJB 提供者(可以对 EJB 组件进行硬编码实现 EJB 组件安全)、应用组装者(为应用提供一组安全角色、为 EJB 方法定义方法许可或是授权)、部署者(将 EJB 部署到 EJB 容器和可操作的 EJB 平台,其次还必须指定哪些主体必须授权给安全角色)、系统管理员(负责安全领域的配置和正在进行的用户和组身份管理:定义用户和用户组,将他们映射到 EJB 安全角色,并且提供恰当的跨越多个企业中或企业间的安全领域的主体映射)、EJB 容器提供者(提高 EJB 容器安全,主要是声明方式的安全性)。Servlet 调用 AccountBean EJB,请求被分派到 EJB。对其进行角色检查。如图 3 中步骤③

所示,当 account servlet 调用 AccountBean EJB 时,Web 容器分派请求到 EJB 容器。请求是通过 IIOP 发送的。因为这个请求是通过安全环境送出的,所以 CSIv2 协议是有效的。支持 Web 容器和 EJB 容器运行的服务器使用 CSIv2 协议建立一个安全关联。在连接成功并对 user1 的证书进行验证后,EJB 容器所接受到的身份是 user1 的。

(2)EJB 容器通过 JAAS 机制来确保电子商务/电子政务系统安全。user2 客户端采取 JAAS 登录,直接通过 IIOP 发出请求调用 AccountBean EJB,此时,用来认证和授权的 J2EE 安全技术是 JAAS(Java 认证和授权服务)客户端使用 javax. security. auth. callback. CallbackHandler 进行配置,并对支持 EJB 容器的服务进行 JAAS 登录(如图 3 步骤⑥所示),当 user2 提供了有效的 ID 和密码对以后,他能够执行 AccountBean EJB 上的方法调用。其 AccountBean EJB 模块的部署描述符中的 XML 元素 method - perssion 示例如下:

```
<method - permission>
<role - name>Teller</role - name>
<method>
<ejb - name>AccountBean></ejb - name>
<method - name>getBalance</method - name>
</method>
</method - permission.
```

(3)任何 EIS 中的信息都必须禁止未授权的访问。EIS 系统一般具有自己的授权模型,JCA 是用来为基于 J2EE 的应用扩展端到端安全模型以整合入 EIS,其 EJB 通过 JCA 标准允许两种登录方式(容器管理登录和组件管理登录)登录 EIS,确保 EIS 被安全地访问,从而确保电子商务/电子政务系统安全,如图 3 的步骤⑦。使用容器安全管理登录,可以通过声明性实现与一个 EIS 的连接,如果一个连接要成为容器管理的,那么部署描述符必须指出,与资源定义相关联的 res - auth(资源授权)元素应该申明为 Container,其部署描述符中的一个 XML res - auth 元素如下:

```
<resource - ref>
<description>Connection to myConnection</description>
<res - ref - name>eis/myConnection. cci. ConnectionFactory</res - type>
<res - auth>Container</res - auth>
</resource - ref>
```

如果连接是通过程序性(组件管理登录)传递身份信息来获取资源信息的,那么 res - auth 的值应该被设为 Application,其部署描述符中的一个 XML res - auth 元素仅须将上述代码中的 Container 改为 Application。

正如在具有强加了安全特性的简单请求流程中所描述的那样,J2EE 安全模型提供了在电子商务环境中进行安全事务操作的能力和基础结构。"

4 结束语

J2EE 平台在构架安全电子商务/电子政务系统有着独特的优势,其安全体系结构亦非常复杂。文中仅从整体上介绍了 J2EE 的安全架构,其 Web 服务的企业级安全、对容器提供者的安全,以及 J2EE 环境下的 PKCS、S/MIME 加密技术安全和 SSL、TLS 协议安全技术都未做探讨。这些安全技术对搭建安全电子商务/电子政务系统亦起着非常重要的作用,仍然是值得研究的领域。

参考文献:

[1] 魏楚元:J2EE 安全机制的分析与研究[J]. 计算机工程与设计,2005,26(26):2 - 5.

[2] 徐迎晓. Java 安全性编程实例[M]. 北京:清华大学出版社,2003.

[3] Pistoia M, Nagaratnam N. 企业级 Java 安全性——构建安全的 J2EE 应用[M]. 尹 亚,明喻卫,严进宝译. 北京:清华大学出版社,2005.

[4] Greenstein M, Feinman T M. Electronic Commerce: Security, Risk Management and Control[M]. 北京:机械工业出版社,2000.

[5] 韩宝明,杜 鹏,刘 华. 电子商务的安全与支付[M]. 北京:人民邮电出版社,2004.

[6] 章评平,汪秉文,戴志诚. 基于 J2EE 平台的电子商务开发[J]. 微机发展,2004,14(9):66 - 68.

(上接第 203 页)

服务器实现了安全代理的集中管理,使得网络安全管理更加智能化、自动化,大大减轻了网络管理员的劳动量。安全代理和服务器之间的加密通信,确保了所有管理工作的的安全性。安全代理的大部分工作都是由位于内核空间的功能模块来完成,可以减少资源利用,增加安全代理本身的安全性。同时,安全代理也支持第三方软件,可以把第三方软件作为它的运行在用户空间的功能模块,这使得安全代理模型有很强的扩展性、通用性。

参考文献:

[1] Wu Pufeng. Intrusion detection with snort[M]. Beijing: China Mechanism Press, 2005.

[2] Caswell B, Beale J, Foster J C, et al. Snort 2.0 Intrusion Detection[M]. 北京:国防工业出版社,2004.

[3] Müller K. Intrusion Detection System[EB/OL]. http://www.linuxfocus.org, 2003.

[4] Hess A, Jung M, Schafer G. A Flexible Intrusion Detection and Response Framework for Active Networks[R]. Kemer - Antalya, Turkey: ISCC, 2003.

[5] Hu Daoyuan, Min Jinghua. Network security[M]. Beijing: Tsinghua University Press, 2004.

[6] Stallings W. Network security essentials applications and standards[M]. 北京:清华大学出版社,2004.