

J2EE 中基于 CAPTCHA 的认证模块的实现

王海龙, 朱程荣

(同济大学 计算机科学与工程系, 上海 200092)

摘要:目前基于 J2EE 平台的信息系统的使用越来越广泛,但是,它在身份认证技术方面还存在一些不足,例如,它易于被自动填表机器人程序利用反复提交表单,从而抢占了网络连接。文中提出了一种基于 CAPTCHA 的认证模块与身份认证技术 JAAS 相结合的实现方式,它扩展了一般的身份认证技术的功能,解决了区分真人和计算机的问题,使认证更安全。最后把该技术应用到了实际的项目中,取得了满意的效果。

关键词:图灵测试;Java 认证和授权服务;J2EE

中图分类号:TP393.08

文献标识码:A

文章编号:1005-3751(2006)01-0228-03

Realization of CAPTCHA-Based Authentication Module for J2EE

WANG Hai-long, ZHU Cheng-rong

(Department of Computer Science and Engineering, Tongji University, Shanghai 200092, China)

Abstract: Although application systems based on J2EE platform is more popular, it is insufficient in ID authentication, such as automatically submitting form by computer programs called Auto-Filling Robot, then net connections are wasted. The article brings forward a method to implement the integrating CAPTCHA with JAAS. We extend the common authentication's function and solve the problem how to tell computers and humans apart by this kind of method that makes authentication more secure. Finally, our project apply the technology, and it works not bad.

Key words: turing test; JAAS; J2EE

0 引言

网络是人们相互交换信息和共享资源的平台,但其中也充斥着大量的垃圾信息,甚至有人用一些自动填表程序专门散布这些垃圾信息,严重干扰了网络信息的正常传播。比如,Blogs 和一些提供下载的网站,如果被人用自动填表机器人软件反复提交表单,这些网站将被毫无价值的信息淹没,网络连接也被抢占;而那些重要的信息也不易被人发现。一种已经被很多网站或网络应用系统(如 Yahoo! 网站)广泛使用的预防技术是 CAPTCHA(Completely Automated Public Turing test to tell Computers and Humans Apart,全自动区分计算机和人类的图灵测试)。

CAPTCHA 用以区分计算机和人类,在人机差别非常小的网络上非常有效。它会生成一个随机的图片显示给用户。这个图片含有一个不容易被计算机识别的字符串,同时这个字符串在页面的代码及其他计算机可以获取的地方被使用。如果表单提交的时候并不含有正确的字符串,系统就能够确信输入人员录入错误或者不是一个真正的人在录入。

1 J2EE 的安全机制

J2EE 的体系结构导致在应用程序中强制进行安全检查。在 J2EE 环境中,组件的安全是由它们各自的容器来负责的,组件的开发人员几乎可以不用或者很少在组件中添加有关安全的代码。这种安全逻辑和业务逻辑相对独立的架构使得企业级应用系统有更好的灵活性和扩展性。J2EE 产品通常为应用程序开发者提供两种形式的基于容器的安全机制:声明性安全和编程性安全^[1]。

1.1 声明性安全

声明性安全通过安全结构描述的方式代表应用程序的安全需求,安全结构一般包括安全角色、访问控制和验证要求等。在 J2EE 平台中部署描述符(web.xml, ejb-jar.xml)充当了声明性安全的主要工具。在程序运行时容器从部署描述符中提取出相应的安全策略,然后容器根据安全策略执行安全验证。声明性安全不需要开发人员编写任何安全相关的代码,一切都是通过配置部署描述符来完成的。

1.2 编程性安全

可编程式安全性在声明性安全性的基础上,使安全敏感的应用可以通过调用容器提供的 API 来对安全做出决断。这在说明性的安全机制不足以满足企业的安全模型的情况是非常有用的。J2EE 在 EJB EjbContext 接口和 Servlet HttpServletRequest 接口中各提供了两个方法:is-

收稿日期:2005-04-29

作者简介:王海龙(1978—),男,湖北人,硕士研究生,研究方向为计算机网络信息安全。

CallerInRole(EJBContext)、getCallerPrincipal(EJBContext)、isUserInRole (HttpServletRequest) 和 getUserPrincipal (HttpServletRequest)。这些方法允许组件根据调用者或远程用户的安全角色来作出商业判断^[2]。

2 Java 验证和授权框架

Java 认证和授权服务 (Java Authentication Authorization Service, JAAS) 为 J2EE 应用程序提供了为一个特定的用户或一组用户进行认证和授权。JAAS 是标准的可插入认证模块 PAM 的 Java 版本, 它对 Java 2 平台的安全认证框架进行了扩展以支持基于用户的安全认证。

通过在应用程序和底层的验证和授权机制间加入一个抽象层, JAAS 可以简化涉及到 Java Security 包的程序开发。抽象层独立于平台的特性使开发人员可以使用各种不同的安全机制, 而且不用修改应用程序级的代码。和其他 Java Security API 相似, JAAS 通过一个可扩展的框架——服务提供者接口 (Service Provider Interface, SPI) 来保证程序独立于安全机制。服务提供者接口是由一组抽象类和接口组成的^[3]。图 1 中给出了 JAAS 程序的概貌。应用程序层的代码只需要和 LoginContext 打交道。在 LoginContext 之下是一组动态配置的 LoginModule 对象, 这些对象使用相关的安全基础结构进行验证操作。

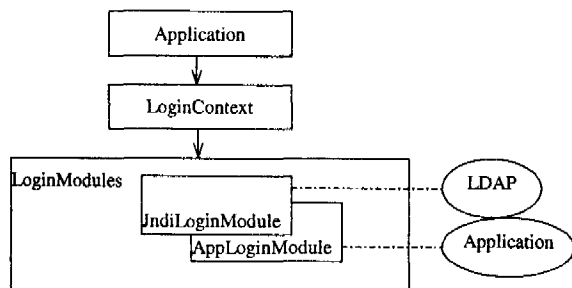


图 1 JAAS 概貌

3 新的认证系统的实现

3.1 设计目标

在分析完 J2EE 的安全服务体系后, 将根据 J2EE 的规范, 在基于 J2EE 的应用中增加基于 CAPTCHA 的认证模块。具体目标如下:

(1) 实现声明性安全认证机制, 因为声明性安全方式更加灵活, 可扩展性好。在改变角色权限的时候只需要改变部署描述符的声明部分即可, 而不需要改动任何的应用代码。

(2) 扩展 JAAS 的 LoginModule 模块, 使之能够同时验证用户名与密码是否相匹配和 CAPTCHA 字符串与用户输入是否相匹配。

(3) 新的认证系统应该能区分人类和计算机输入, 防止自动填表程序自动输入。

(4) 能够方便地移植到其他的基于 J2EE 应用系统中。

3.2 解决方案

3.2.1 声明要保护的资源

在应用程序部署描述文件 (web.xml) 中分别对 <web-resource-collection>, <auth-constraint>, <user-data-constraint> 元素定义, 分别实现对要保护资源的声明, 及角色和传输协议的限制。并增加 <login-config> 元素定义, 实现用户提交后的页面跳转^[4]。

3.2.2 产生 CAPTCHA 图片

JSP 中对 CAPTCHA 模块的调用是通过 标签的 src 属性值来指定实现的; 实际上, src 属性也是对资源的请求, 但是这个资源不是静态的, 而是一个 Servlet, 这个 Servlet 已经在 web.xml 中指定好了。然后, 这个 Servlet 在执行过程中调用 CAPTCHA 模块产生随机图片, 最后发送到客户端。

JSP 中代码片段如下:

```
<%产生一个 CAPTCHA 图片%>
```

```

```

为了更加灵活地使用不同的 CAPTCHA 模块, 在这里使用了抽象工厂的设计模式。需要使用哪一种 CAPTCHA, 只需要在配置文件中设置即可^[5]。Jcaptcha 是一个开源的实现了 CAPTCHA 功能的 Java 项目 (其网站为 <http://jcaptcha.sourceforge.net>), 它是一个比较专业的能够产生带有验证字符的图片的 CAPTCHA 模块。而下面这段代码也是一个简单实现框架, 它也能产生带验证码的图片:

```
public class SimpleToken implements Token {
    protected String authString = null;
    public SimpleToken(String token) { this.authString = token; }
    public BufferedImage getTokenImage() { }
    public boolean validateResponse(Object response) { }
}
```

getTokenImage() 方法对外提供图片, 它内部实现主要是用产生随机数字, 也可以将这些数字转化为英文字母, 然后应用 java.awt 包的类把这些字符“写”到图片中去, 并产生一些特别的视觉效果 (干扰自动填表程序读取字符)。validateResponse() 方法用来检验用户输入的验证码是否正确。

3.2.3 扩展 JAAS login module

在这一步中, 需要实现 JAAS LoginModule 接口, 并涉及到以下几个步骤:

(1) 创建一个 LoginContext 的实例。

(2) 为了能够获得和处理验证信息, 将一个 CallbackHandler 对象作为参数传送给 LoginContext。

(3) 通过调用 LoginContext 的 login() 方法来进行验证。

在 login() 方法中, 可以从 CallbackHandler 对象获取用户的输入 (包括用户名、密码和验证码), 然后同时检查用户名与密码是否匹配, 验证码是否正确。

3.2.4 集成到应用服务器

最后需要把整个认证模块集成到应用服务器中去。目前在应用服务器市场中的 JAAS 产品还不是很一致,使用 JAAS 的 J2EE 应用服务器有一些细微的差别。例如 JBoss 使用自己的结构,将 JAAS 集成到了一个更大的安全框架中;而虽然 WebLogic 也使用了 JAAS,安全框架却完全不一样。在文中所涉及的应用中,使用的是 Tomcat 5。部署在该服务器中的应用程序需要在它的部署描述文件中增加<Realm>元素的定义:

```
<? xml version = '1.0' encoding = 'utf-8'? >
<Context docBase = "d:/captcha - login/web" path = "/login" debug = "9">
< Realm className = "org.apache.catalina.realm.JAASRealm"
appName = "clogin" debug = "99" roleClassNames = "ww.jaas.CPrincipal"
userClassNames = "ww.jaas.CPrincipal"/>
</Context>
```

将认证模块插入到应用中去,仅仅需要的是一个配置文件,如下代码段所示:

```
clogin|ww.jaas.Clogin Module required debug = true;
};//JAAS 配置文件:jaas.config
```

3.2.5 新的认证模块的实际应用

在一个基于 J2EE 平台的房地产交易系统中,使用了该认证模块,有效防止了机器自动登录,阻止了垃圾信息的发布,增强了安全性。其运行效果如图 2 所示。

4 结束语

为避免人为利用自动填表程序反复登录或发表垃圾信息,许多基于 J2EE 平台的网络应用系统必须区分是真正的人还是自动填表程序在操作。CAPTCHA 能够在人和自动填表程序之间作出判断。在文中,介绍了 CAPTCHA 的相关概念,并对 J2EE 中安全策略和 Java 认

证和授权服务作了一些阐述;最后在这些概念基础上,用 CAPTCHA 扩展了 JAAS 中的登录模块,并把这个模块集成到应用服务器中。由于上述实现过程是建立在标准 J2EE 安全机制上,所以可以在任何 J2EE 应用上复用。

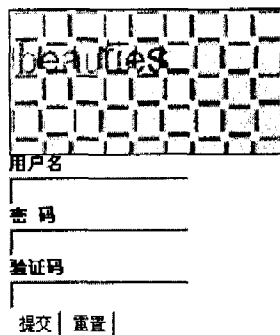


图 2 运行效果

参考文献:

- [1] 王 敏,吉 逸. Java2 环境下身份认证和授权机制的研究[J]. 微机发展,2003,13(5): 40-42.
- [2] 赵仲孟,沈海斌. J2EE 应用服务器安全服务体系的分析与实现[J]. 计算机工程与应用,2003,39(21): 175-177.
- [3] Sun Microsystems. Java™ Authentication and Authorization Service (JAAS) LoginModule Developer's Guide[EB/OL]. <http://java.sun.com/j2se/1.4.2/docs/guide/security/jaas/JAASRefGuide.html>,2001.
- [4] Sun Microsystems. Java™ Authentication and Authorization Service (JAAS) Reference Guide for the Java2 SDK, Standard Edition[EB/OL]. <http://java.sun.com/j2se/1.4.2/docs/guide/security/jaas/JAASLMDevGuide.html>,2001.
- [5] von Ahn L, Blum M, Hopper N J, et al. CAPTCHA: Using Hard AI Problems For Security[J]. Lecture Notes in Computer Science, 2003, 2656(9): 294-311.

(上接第 227 页)

分析表 1 中 NBC 与 C4.5 在入侵检测中性能的比较,可以看出朴素贝叶斯分类器的准确率同决策树的准确率相当,但其无论是学习还是分类时间都明显低于 C4.5 的时间。这是因为,朴素贝叶斯的属性独立的假设简化了运算。由于在入侵检测系统中,不仅要求能准确地检测出入侵行为,而且还需要快速检测出入侵行为,这样才可能及时采取措施,降低或消除入侵行为对系统造成的破坏。通过实验可以证实朴素贝叶斯分类器可以满足入侵检测的这种要求。由于朴素贝叶斯分类器的前提是属性的严格独立,在众多的约简算法中,基于粗糙集的约简可以实现这种约简。

但是由于朴素贝叶斯分类器的理论基础是贝叶斯法则,当测试数据同训练数据的概率分布出现比较大的差异时,朴素贝叶斯分类器的准确率会受到严重影响。所以将朴素贝叶斯分类器用于网络入侵检测系统,要求经常更新

训练数据,保证训练数据与网络的实时环境中的网络行为具有基本一致的概率分布。

参考文献:

- [1] Mitchell T M. 机器学习[M]. 曾华军,张银奎,等译. 北京:机械工业出版社,2003.
- [2] Information and Computer Science University of California, Irvine. KDD cup 1999 Data[EB/OL]. <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>,1999-10.
- [3] 刘 清. Rough 集及 Rough 推理[M]. 北京:科学出版社,2001.
- [4] 史忠植. 知识发现[M]. 北京:清华大学出版社,2002.
- [5] Knowledge Systems Group, Dept. of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway. ROSETTA[EB/OL]. <http://rosetta.lcb.uu.se/general/download/1999>.