Java开发专家

# 敏捷Acegi、CAS
## ——构建安全的Java系统

零距离接触源于Spring血统的Acegi

揭露CAS单点登录服务器的使用方法及内部架构

深入到Acegi、CAS SSO源码中

全面跟进Acegi、CAS、Java EE安全性编程模型的最新技术

罗时飞 编著
飞思科技产品研发中心 监制

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

Java开发专家

# 敏捷Acegi、CAS

## ——构建安全的Java系统

罗时飞 编著
飞思科技产品研发中心 监制

電子工業出版社
Publishing House of Electronics Industry
北京·BEIJING

# 内 容 简 介

本书是关于 Acegi、CAS 的权威教程，是 Java/Java EE 安全性开发者的必备参考书。无论是 Java EE 安全性编程模型的背景和基础知识，还是 Acegi、CAS 本身，还是有关 Acegi、CAS 的各种高级使用技巧和最佳实践，本书都详尽、系统地给出了阐述。全书共分为 4 部分：第 1 部分介绍 Web 应用安全，主要围绕 Java EE 安全性编程模型、从宏观上看待 Acegi 及其初步使用等方面进行阐述；第 2 部分介绍 Acegi 认证支持，主要围绕 Acegi 支持的各种认证机制、各种认证提供者、各种企业级特性等内容展开论述，还重点介绍了 Acegi 内置的 Captcha 集成支持、Java EE 容器适配器支持等；第 3 部分介绍 Acegi 授权支持，主要围绕 Web 资源、业务方法、领域对象的授权操作展开论述；第 4 部分介绍 CAS 3 认证支持，它主要从 CAS 3 服务器的使用及其内部架构角度给出论述。全书理论与实践并重，通过大量的实例帮助读者尽快掌握 Acegi 的使用技巧，从而提高本书的参考、阅读价值。

本书适合作为 Java/Java EE 安全性开发者、系统分析师和架构师的参考书，同时，本书非常适合于高校相关专业的学生，以及对 Java/Java EE 安全性有兴趣的各类开发者。



**图书在版编目（CIP）数据**

敏捷 Acegi、CAS：构建安全的 Java 系统 / 罗时飞编著. —北京：电子工业出版社，2007.4
（Java 开发专家）
ISBN 978-7-121-03888-4

Ⅰ. 敏… Ⅱ.罗… Ⅲ.Java 语言－程序设计 Ⅳ.TP312

中国版本图书馆 CIP 数据核字（2007）第 022810 号

责任编辑：王树伟
印　　刷：北京市通州大中印刷厂
装　　订：三河市鹏成印业有限公司
出版发行：电子工业出版社
　　北京海淀区万寿路 173 信箱　邮编：100036
开　　本：787×1092　1/16　印张：35.5　字数：908.8 千字
印　　次：2007 年 4 月第 1 次印刷
印　　数：6 000 册　定价：59.80 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系电话：（010）68279077；邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

“开发专家之 **Sun ONE**”全新提升为“**Java** 开发专家”系列
——源自精品 成就理想

# 出 版 说 明

## ★ 从“开发专家之 Sun ONE”到“Java 开发专家”

“开发专家之 Sun ONE”系列丛书从诞生之日至今，已经四岁了。在这个系列里面，我们一直努力体现着这么一个理念：用一种较为敏锐的视角来跟踪 IT 技术的发展轨迹，并把可能为广大程序员所希望获得的知识，用图书出版的方式奉献给大家。

在这个系列中，我们陆续出版了约 30 种图书，有《Java 与模式》、《JSP 应用开发详解（第 2 版）》、《精通 EJB（第 3 版）》、《Tomcat 与 Java Web 应用开发详解》、《精通 Struts：基于 MVC 的 Java Web 设计与开发》、《JBoss 管理与开发核心技术（第 3 版）》、《精通 Spring》、《精通 Hibernate：Java 对象持久化技术详解》等一大批读者朋友耳熟能详的作品。很多作品都是在国内没有同类图书的情况下出版的。在这几年的出版工作中，我们时刻感受着市场的风险，也时刻收获着无数读者给我们的认可。

在这个系列中，凝聚了大量资深技术专家的心血。有大家都熟知的阎宏、刘晓华、孙卫琴、罗时飞等，还有一些正在不断腾跃的开发高手。这些非常优秀的国内原创作者们一直都在支持着“开发专家之 Sun ONE”系列的出版工作，在这里，我们要向他们说声：谢谢。

桃李不言，下自成蹊。由于这些年“开发专家之 Sun ONE”在“两个效益”中的杰出表现，电子工业出版社授予这个系列“最佳品牌奖”。

时代不断前进，技术不断变革。为了顺应 Java 领域的技术发展态势，为了赋予这个经典的图书系列更强的生命力，我们将“开发专家之 Sun ONE”升级为“Java 开发专家”。我们将继承原有的出版理念，紧密跟踪技术热点和发展趋势，会聚更多优秀作者，全力奉献更经典的作品。

## ★ 规划你的 Java 开发之路

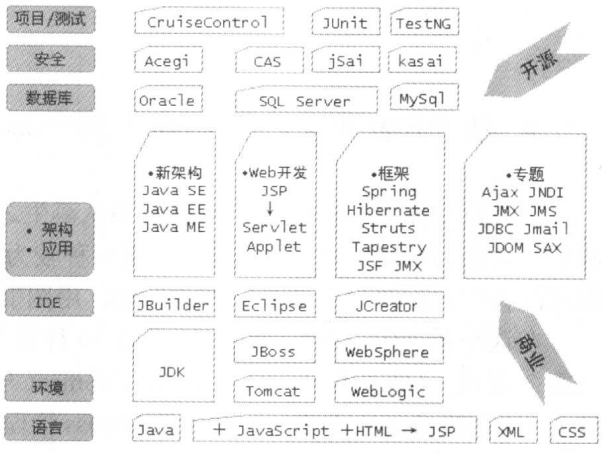
喜马拉雅山脉的最高峰不断地在温室效应中降低，而 Java 世界的颠峰永远都在技术人员的追求中不断升高。每个人都有不同的路，每个人都有不同的行路方式，不过，往往“到了山顶才发现，错误的路和正确的路就差那么几步！”

身处 Java 洪流中的程序员最累（不过大家都说 Java 程序员薪水最高，呵呵），我们简单整理了一下 Java 领域的相关技术、工具、架构，如下图所示。这个框图中的每一个英文单词（或缩写）都可以写成一本书。Java 领域还有一个特点，那就是商业产品和开源产品层出不穷，潮流不断。相比于其他领域，如.NET，Java 开发更是体现了这句谚语：条条大路通罗马。

罗马只有一个，大路却有多条。看上去，似乎到罗马很容易，反正路多嘛。不过，路多却容易迷失方向。当你在 Java 领域中摸爬滚打几年后，发现自己在无数条道路上走了很久，却不知道罗马何日才能到达，甚至连罗马的方向都不知道，这时你肯定会很失落。

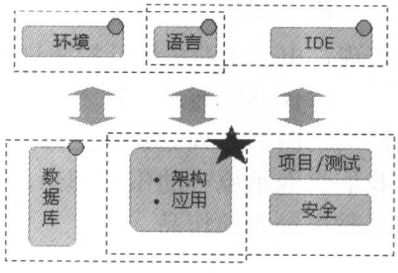
很遗憾，在这个简短的出版说明文章里面，我们无法告诉你每一条连贯的、不费周折的通往罗马的道路该如何走。或许，通过“Java 开发专家”系列中的某本书，你可以找到属于你的正确道路。在一般情况下，我们不会就某一项很窄的话题来单独写一本书，我们还是希望通过我们的一些专业和智慧，尽力把一些相关技术整合起来，用较为简明的方式表达出来，最后由你来选择。

这里有句话与大家共勉：少走弯路，就是捷径！




## ★ “Java 开发专家”的奉献

犹如在上面那个框图中展现的那样，我们希望在各个层面、各个方向上都能给读者奉献出优秀的图书作品，全面体现技术与应用的结合。从宏观上看，我们会从语言、IDE、环境、数据库、架构与应用、安全、项目与测试等方面进行选择，选出一些读者迫切需要的技术来先行规划。




“Java 开发专家”虽然新蓓初绽，但因其源自盛放的“开发专家之 Sun ONE”系列而根基稳健，两个系列会有一段很长的并行时间，我们会用一种优化的方式来保证读者的顺利选择。无论哪一个系列，必定都有大家喜欢的图书。

在技术上，有着持久化的方法，在学习上，也需要有持久化的精神。
从“开发专家之 Sun ONE”到“Java 开发专家”，希望可以带给你持久化的动力。

联系方式

咨询电话：（010）68134545　　88254160
电子邮件：support@fecit.com.cn
服务网址：http://www.fecit.com.cn　　http://www.fecit.net
通用网址：计算机图书、飞思、飞思教育、飞思科技、FECIT

JA-SIG Central Authentication Service (CAS) is one of the leading open source enterprise-level single sign on solutions available today. CAS is developed under the auspices of JA-SIG, a non-profit organization consisting of institutions of higher education and commercial companies with the goal of fostering collaboration amongst institutions around the world to develop software and share knowledge. CAS is maintained and enhanced by a core group of developers from around the world including the United States, France and Belgium.

To people, CAS means many things. CAS is a protocol, server, and collection of client libraries. The CAS server is an open-source J2EE application that users can download and deploy. The client libraries come in many flavors: Java, PL/SQL, .NET, PHP, Perl, Ruby, Python, and more. These libraries allow applications to communicate with the CAS server over the CAS protocols in order to participate in single sign on.

CAS, including its original protocols, was developed by Yale University. The original protocol allowed untrusted web applications to authenticate users against a trusted central server. A typical scenario would involve a client, typically a web browser, an application (the CAS client), and a CAS server instance. When a user attempted to access a secure application, his web browser would be redirect to the CAS server for authentication. The CAS server would utilize a backend data store such as LDAP or a database to validate the user's credentials. Upon successful authentication, the CAS server would generate a ticket, and redirect the browser back to the application, appending the ticket to the application's URL. The application would then communicate to the CAS server over a secure channel and validate the ticket. Upon successful validation, the CAS server would return the user's ID to the application.

Later versions of the CAS protocol added features such as an XML response and multi-tiered proxy authentication. Multi-tiered proxy authentication is essential for the portal deployments. It allows portal to access content on remote servers and applications without having to store and "re-play" a user's password. In addition, the CAS protocol supports authentication across domains, something most other single sign on solutions do not support without the usage of federation.

Yale developed the CAS Server 1.0 and 2.0. However, in December of 2004, Yale joined forces with Rutgers University and jointly developed and promoted the CAS Server 3.0 under the JA-SIG umbrella. While the newest CAS Server did not advance the protocol, it was a completely re-architected server application, with a focus on future development. Built on open source libraries like the Spring Framework, Spring Web Flow, xFire, and numerous Apache libraries, CAS Server 3.0 is extremely flexible and provides a completely pluggable architecture. This architecture allows local developers to completely customize the server without worrying about "forking" the code base. It also allows the CAS server team to quickly add new features while easily maintaining backwards compatibility. Because of this, the standard CAS Server release now includes plug-ins for connecting to databases, files, LDAP servers, X.509 certificates, and RADIUS servers as well as advanced support for server clustering.

Now that the CAS server has a stable and mature server architecture, the CAS team is looking to introduce more enhancements and support additional protocols. The planned CAS Server 3.1 release will support SAML 1.1 and SAML 2.0, which will improve interoperability between CAS and other single sign on solutions. In addition, more robust services management will be included. This will allow deployers to specify which services are able to use CAS, and which features of CAS they are allowed access to! Finally, the CAS server will also support supplying common user attributes to applications requesting authentication. CAS will also ship with additional plug-ins such as SPNEGO support. JA-SIG will be working with the client library owners to ensure support for these new CAS features.

As CAS has grown into a mature, robust, extensible project, its adoption and usage has also grown. CAS is deployed worldwide including the United States, China, Sweden, France, Canada, Spain, United Kingdom, Colombia, Greece, India, Romania, Russia, Belgium, Netherlands, Australia, and Japan. CAS is deployed by a wide-range of higher education institutions, commercial entities, non-profit organizations, and government agencies. The largest CAS deployment currently has five hundred thousand active users, while many average over fifty thousand active users. CAS has been used to secure applications such as email clients, financial packages, portals, Human Resources information, online games, student grades, and more.

CAS has become ubiquitous in the security world. In addition to the client libraries detailed before, support for CAS is built into numerous applications including JA-SIG HyperContent, JA-SIG uPortal, Mantis, TikiWiki, Mule, Moodle, LifeRay and the BlueSocket wireless access points. Many of these applications (and many others not listed) are able to easily include CAS support due to their usage of Acegi Security for Spring (sometimes called Spring Security). Acegi, which provides declarative authentication and authorization capabilities to Spring applications, has included support for CAS since its inception. Acegi supports all of the major CAS features including proxying and opting out of single-sign on. CAS can be added to any application utilizing Acegi by merely updating the Acegi configuration file to use the authentication filters designed to support the CAS protocol.

If you're looking for more information, the following web resources provide excellent background information on CAS, JA-SIG and Acegi:

- JA-SIG Web Site: http://www.ja-sig.org
- CAS Web Site: http://www.ja-sig.org/products/cas/
- Acegi Security Web Site: http://www.acegisecurity.org

Scott Battaglia

## 奔向罗马

罗马，素有“永恒之城”的美称。如果将这一美称用于开源领域的 Acegi、CAS 身上，则一点也不为过。从 Java EE 平台技术的发展、变化来看，Java EE 中安全性的使用非常混乱，各个容器厂商都以这样或那样的方式开发并架构自身的 Java EE 安全性基础框架，最终将它们所倡导的 Java EE 安全性编程模型强加在开发者身上。Acegi 的出现改变了原有的、混乱的格局。同样地，迄今为止，现有的 JCP 组织并没有给出同单点登录相关的任何 Java EE 规范。相比之下，一直以来，历史悠久、著名的 CAS 中央认证服务在单点登录领域占据了非常重要的地位。Acegi 与 CAS 在一同护卫企业级应用，它们专注于 Java EE 平台中的安全性技术，并提供了卓越的解决方案。

多年前，作者所架构的许多平台技术需要同时支持各种 Java EE 容器，这些平台技术几乎与特定 Java EE 容器无关。这些平台技术使用到大量 Java EE 安全性技术。当 Acegi 出现在作者的视野时，其优美的架构、灵活的部署策略、稳定的基代码，加上它提供了 Java EE 安全性编程模型未提供的各种企业级特性等，这些特性深深地打动了作者。原生的 Java EE 安全性编程模型确实不错，但是企业、开发者在实施基于这一原生模型的企业应用时，他们到处碰壁，因为各个 Java EE 容器采取了不同的策略和方式暴露各自的 Java EE 安全性编程模型。我们需要更便携地架构、开发和实施 Java EE 应用，这在各 Java EE 容器趋于同质化的今天尤为重要。正如 Spring 看到 Java EE 编程模型存在的缺陷一样，Acegi 看到了 Java EE 安全性编程模型所存在的各种缺陷。

便携性是优秀软件项目、技术框架必须具备的基本特质之一。当然，便携性不是 Acegi 的唯一卖点，这也不是作者（包括你）看重 Acegi 的唯一理由，正如理解和掌握各种设计模式是开发者的必修课一样。同样的 Acegi 使能应用能够部署到任意场合、任意 Java EE 容器。当便携性成为 Acegi 的“性格”的组成部分时，开发者便能够在各种场合一劳永逸地享受到 Acegi 提供的 Java EE 安全性支持。开发者再也不用依据特定 Java EE 容器的“脸色”行事。

Acegi 能够保护 Web 资源、业务方法及领域对象，而标准 Java EE 安全性编程模型不能够保护领域对象。为借助于 Acegi 保护这些资源，在 Acegi 实现内部，大量的最佳实践被应用到其中，比如采用过滤器链和 AOP 拦截器链拦截客户请求、领域驱动设计的应用、策略架构模式、观察者模式等。从应用开发和 Acegi 的使用角度考虑，开发者只需要借助于 Spring 配置文件配置同 Acegi 相关的所有受管 Bean。很显然，Acegi 使能应用的测试工作能够顺利展开，因为 Acegi 是基于 Spring 架构开发而成的。甚至，在 Acegi 使能应用的开发期间，测试驱动开发（TDD）的实施效果非常好。敏捷的 Acegi 为企业应用的快速、高质量开发奠定了非常坚实的基础。

当我们借助于 Acegi 或其他安全性技术部署了若干企业应用时，这些应用自身都需要实现用户的认证逻辑。此时，为登录到不同的应用，用户必须提供不同的用户账号。可以看出，这时的应用认证逻辑并未得到重用，而且用户自己需要维护非常多的账号。借助于单点登录解决方案，企业、开发者能够将所有的认证逻辑委派给 SSO 产品，比如 JA-SIG CAS 服务器。

中央认证服务（CAS）是一个著名的 SSO 产品，各个应用可以将用户认证请求委派给它。CAS 服务器部署方便、安全、稳定，而且开发者能够轻易地自定义它，比如提供本地版本的登录界面、调整 SSO 认证逻辑、自定义登录流程等。各个 CAS 服务器都严格遵循 CAS 1.0 和 2.0 协议，这些协议定义了各个 CAS 服务器必须实现的语义。尤其需要特别指出的是，CAS 2.0 协议引入的代理特性使得实现了 CAS 2.0 协议的各个 CAS 服务器适合于多层部署环境，这无疑拓展了 CAS 服务器的应用领域，比如基于 SOA（面向服务架构）、ESB（企业服务总线）的企业计算环境。

无论是 Acegi，还是 CAS，它们收集用户凭证的策略和方式非常灵活。比如，它们都允许用户使用 X.509 CA 客户证书作为登录凭证；它们都允许开发者采用 LDAP、RDBMS 等存储源存储用户信息；它们都提供了一流的 JAAS 集成支持；它们同时支持远程客户。Acegi 同时支持 HTTP BASIC、表单、Digest、CLIENT-CERT 等认证机制，它甚至提供了一流的 Captcha 集成支持。如果需要，开发者还可以使用 Acegi 内置的 Java EE 容器适配器，这使得 Acegi 与 Java EE 安全性编程模型能够出现在同一企业应用中。Acegi 使能应用中的安全性上下文能够透明地传播到各处，看来，我们能够将 Acegi 应用到任何企业计算环境中。

尽管 Acegi 与 CAS 都在各自的安全性领域摆出了领先者的姿态，但它们的协同工作能力也是一流的。比如，开发者可以在 CAS 服务器中使用到 Acegi 提供的集成支持，而 Acegi 使能应用也可以将认证请求委派给 CAS 服务器，从而使得用户能够享受到单点登录的乐趣及 CAS 服务器带来的便利。

今天，大量基于 Acegi、CAS 的企业系统已经在生产环境中稳健地运行，作者有幸架构和实施了若干个这类大型系统。值得重点一提的是，即使开发者不打算在自身的应用中启用 Acegi、CAS，但借助于它们去熟悉 Java EE 安全性编程模型不愧是一种绝佳选择，因为熟悉 Acegi、CAS 的门槛并不高。在熟悉 Acegi、CAS 期间，开发者还能够获得保护企业应用一手的最佳实践。让我们开始奔向永恒之城。

在本书出版之际，作者要感谢出版社、读者和家人的大力支持。与此同时，作者要特别感谢 Acegi 开发团队中的 Ben Alex、JA-SIG CAS 开发团队中的 Scott Battaglia，没有他们的帮助，本书会减色不少。尤其是，要感谢 Scott Battaglia 为本书写的英文序。

Java EE 平台所建立的安全性知识体系非常庞大，而写作这样一本专门“针对 Java EE 安全性架构、开发及部署，尤其是 Acegi 和 CAS”的图书也是非常大的一个挑战。尽管作者在这一领域摸爬滚打了多年，但本书还是可能存在不足之处，甚至错误，敬请广大读者批评指正。如果读者有什么好的同 Java EE 安全性相关的解决方案，则不妨同作者分享。

Good Luck in Your Endeavors!

罗时飞
2007 年于广州

网站介绍

**关于 www.open-v.com**

我们专注于 Java EE 平台、敏捷方法及开源技术咨询工作。图书创作能够将我们的咨询经验带给整个社区，我们也希望在不远的将来，open-v.com 能够成为图书创作的重要平台。有特色的技术培训能够缩短开发者、开发团队采纳敏捷技术、方法的时间，因为我们的培训注重理念、学习方法和学习能力的提高。有价值的咨询服务也是我们的重点，我们非常注重咨询的效果，从而真正为企业带来实实在在的价值。

Acegi 是一个著名的 Java EE 安全性框架，CAS 服务器是 SSO 领域著名的软件产品。它们在各自领域都摆出了领先者的姿态，而且它们的协同工作能力堪称一流。

针对用户认证，Acegi 提供了 HTTP BASIC、HTTP FORM、HTTP DIGEST、HTTP CLIENT-CERT 认证支持；针对用户授权，Acegi 提供了 Web 资源的授权、业务方法的授权、领域对象的授权支持。一旦 Acegi 内置的用户认证支持不能够满足企业应用的需求时，开发者可以考虑采用 CAS 服务器。甚至，Acegi 还为企业应用的开发提供了各种有利的企业级特性，比如匿名认证、Run-As 认证服务、Remember-Me 认证服务、退出服务、HttpSession 的并发控制等。如果需要，Acegi 使能应用还可以使用到 Captcha 集成支持、JAAS 集成支持。如果希望在同一 Web 应用中同时启用 Java EE 安全性编程模型和 Acegi，则 Acegi 内置的 Java EE 容器适配器能够满足这一需求。与此同时，开发者可以将用户敏感信息存储到各种存储源中，比如属性文件、XML 文件、RDBMS、LDAP、X.509 CA 证书等。

针对单点登录需求，开发者可以启用 JA-SIG CAS 3 服务器，这是一个协议公开、实现源代码公开、部署方便、安全、稳定、自定义能力强的 SSO 解决方案。Acegi 本身的基代码提供了 CAS 集成支持。

本书正是围绕 Acegi、CAS 中的上述各项内容而准备的。

## 本书特点

- 本书借助于敏捷的 Acegi、CAS，来重点阐述 Java EE 安全性涉及到的各种技术，而且，Acegi 还为企业应用提供了各种企业级特性。凡是有利于企业应用的安全性技术，本书都尽量呈现。
- 尽量将 Acegi 最实用的、动人的一面展现给读者。Java EE 安全涉及到的知识体系非常庞大，Acegi 清晰地将这一知识体系中的主要内容流露出来，这些内容体现在本书中。
- 深入到 Acegi、CAS 的源码当中，以获得一手的 Java EE 安全性知识。通过这些源码，开发者能够了解到它们是如何解决 Java EE 安全性编程模型中现存的各种缺陷的。
- 在写作过程中，注重理论与实践知识并重。本书非常注重基础知识的阐述。与此同时，各章内容采用的示例都是单独的、自成一体的经典 Eclipse 项目。
- 在代码示例的选材上，力求经典、权威。
- 尽量保证书中图、表的清晰和正确性，以提升阅读体验。
- 无论是知识体系，还是写作风格，各章内容统一、自成一体，开发者阅读起来非常舒服。
- 作者尽量将自身架构和开发大型 Java EE/Acegi 使能项目的经验、在从事 Java EE 咨询工作期间获得的 Acegi 高级技巧和最佳实践体现在书中。
- JA-SIG CAS 开发团队的 Scott Battaglia 特意为本书作序。透过这一英文序，读者能够对 CAS 的发展历程有非常清醒的认识。
- 不断改进图书内容。如今，图书创作是作者的主要工作内容之一，因此自身有更大的责任、更多时间完善此书。让我们共同期待第二版的出现。

## 全书内容安排

全书共分为 4 个不同部分：Web 应用安全、Acegi 认证支持、Acegi 授权支持、CAS 3 认证支持。第 1～3 章构成了 Web 应用安全，即第 1 部分内容；第 4～10 章构成了 Acegi 认证支持，即第 2 部分内容；第 11～14 章构成了 Acegi 授权支持，即第 3 部分内容；第 15～16 章构成了 CAS 3 认证支持，即第 4 部分内容。

## 服务网站

针对书中展示的各种 Java 代码、Ant build.xml 和其他脚本，我们特别提供了 Web 网站（http://www.open-v.com）支持。同时，为保证图书同 Acegi（Spring Security）最新发布版的同步，我们会时常更新图书中的源代码，并公布到这一 Web 网站中，欢迎广大读者下载使用。

罗时飞

2007 年于广州

## 第 1 部分 Web 应用安全



## 第 2 部分 Acegi 认证支持

# 第 1 部分

# Web 应用安全

寒冬！寒冬！Java EE 安全性编程模型的寒冬终于被化解了。这全是因为 Acegi 和 CAS 的出现。

在企业计算环境中，安全性从来都是不能够被忽视的话题。从一开始，Java/Java EE 就没有忽视过安全性，这说明 Java EE 天生就是针对企业计算环境而来的。Java EE 提供了卓越的、庞大的安全性基础架构和知识体系，并实现了它们。凡是安全涉及的领域，Java EE 都会跟进。但不幸的是，Java EE 暴露给使用者（开发者）的 Java EE 安全性编程模型却未得到开发者的青睐。尤其是在需要细粒度控制安全性的那些场合，原生的 Java EE 安全性编程模型几乎是“门外汉”。比如，保护领域对象、将资源授权需求存储到 RDBMS 中、动态调整资源授权需求等。最令开发者厌恶的是，各种有关 Java EE 容器安全性配置的问题都抛给了他们。

相比之下，敏捷的 Acegi 打破了这一僵局。所有同安全性相关的配置不用再同特定 Java EE 容器绑定在一起了，因为 Acegi 相关的配置同 Java EE 应用本身绑定在一起。现在，借助于 Acegi 开发安全性应用，开发者再也不用担心应用的测试、部署问题了。Acegi 架构在 Spring 基础之上，而 Spring 架构在 Java EE 平台技术之上，这一层层的抽象确保了 Acegi 能够适合于任何计算平台、任何 Java EE 容器、任何桌面应用。

JA-SIG CAS 服务器是专门针对 SSO 的解决方案，或者将其称为单点登录产品，它有效地填补了 Java EE 平台技术一直以来的空缺。

本书正是围绕 Java EE 安全性技术展开的，并以 Acegi 和 CAS 为中心。借助于 Acegi 和 CAS，开发者能够构建出安全的 Java 系统。全书分为 4 个不同部分：Web 应用安全、Acegi 认证支持、Acegi 授权支持、CAS 3 认证支持。现在开始进入第 1 部分内容——Web 应用安全。

本部分包含如下内容。