一个软件服务协同中的信任管理框架设计*)

徐 锋 曹 春 郑 玮 冯扬悦 吕 建
(南京大学计算机软件新技术国家重点实验室 南京210093)
(南京大学计算机软件研究所 南京210093)

Design of a Trust Management Framework in Software Service Coordination

XU Feng CAO Chun ZHENG Wei FENG Yang-Yue LU Jian
(State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093)
(Institute of Computer Software, Nanjing University, Nanjing 210093)

Abstract Internet-based Web application systems are gradually built as software service coordination systems. In an open, dynamic and collaborative application environment, traditional methods assumed with closeness, centralization and independence are not able to cope with these security problems efficiently. Trust management is a new method for dealing with security issues of open, distributed network application system. However, the traditional policy-based trust management systems have some shortcomings, i.e. complex in policy making, unable to deal with negative security credentials, etc. So, we design a trust management framework in combination with subjective trust model for software service coordination and security decision in Internet environment. This trust management framework has characteristics of operability, reasonability, and flexibility in policy setting.

Keywords Trust management, Trust valuation

1 引言

目前,以 Web 服务为代表的软件服务及软件服务协同已成为一种新兴的 Web 应用形态。随着网络服务的快速增长,Web 应用愈来愈多地表现为由多个软件服务组成的高度动态和开放的协同系统,其在安全需求方面具有了一些新的特点和需求:1) 安全分析主体的复杂化[1],软件服务协同系统安全分析的主体除了最终用户和系统所有者外,还应包括新引入的服务提供者、服务营运者等;2) 安全信息的不完整性[2],服务的本质在于开放,因此不能期望服务使用方与服务提供方之间相互熟知,并完全掌握对方的安全相关信息;3) 安全度量的相对性[3],软件服务协同环境的开放、动态特性,产生了更多影响系统安全的不确定因素,无法使用绝对的、精确的标准来衡量系统的安全性;4) 安全需求的个性化,不同的应用系统和服务对安全的需求不同,即使是组成同一应用系统的服务也具有不同的安全需求;5) 安全措施实施的自适性,软件服务协同环境是高度动态和多变的,安全措施的实施必须能够实时地适应这种变化。一些传统的安全技术和机制,如访问控制列表、X.509公钥证书体系等难以或根本无法满足 Internet 环境下软件服务协同系统对安全的需求。

信任管理提出了一个适宜的解决软件服务协同系统中安全问题的框架,采用第三方签发的安全凭证和本地安全策略,验证请求是否许可。但当前几个已实现的信任管理系统[4~6]均为基于策略的信任管理系统,部分满足了软件协同系统的的安全需求,仍存在如下不足:1) 安全分析主体单一,仅考虑服务方的安全保护,没有考虑服务调用方的安全问题;2) 安全度量的绝对化,采用策略一致性证明验证的方法进行安全度量和决策,该方法过于精确,不能很好适应 Web 安全环境的多变性和不确定性;3) 无法实时满足动态变化的安全环境,安全策略验证的能力和效率有限,并且大部分信任管理系统,在策略一致性证明验证前必须收集足够的安全凭证。此外,安全策略的制定过程较为繁复,并且需要掌握一定的专业知识。

另一方面,一些主观信任模型提出了信任的度量和评估的概念,能较好地反映软件服务协同环境的多变性和不确定性,以及安全信息的不完整性,并且该方法较适合信任信息收集、评估的自动化实现。但当前几个代表性的信任度评估模型[7,8]还存在如下问题:1) 信任的表述和度量的合理性有待于进一步解释,模型倾向于采用事件概率的方式来表述和度量信任关系,均基于一定概率分布假设;2) 不能很好地解决恶意推荐对信任度评估的影响,现有模型大多采用求简单算术平均的方法综合多个不同推荐路径的信任度;3)缺少灵活的机制,如参数设置,来反映不同实体进行信任评估时所具有的个性特点。

主观信任度模型与传统的基于策略的信任管理系统相结合是一种较自然的解决 Internet 环境中软件服务协同系统安全问题的途径。为此,我们提出了一个基于经验的信任度计算模型,并在此基础上设计了一个基于信任度的信任管理框架。该信任管理框架具有易操作性、合理性和策略设置的灵活性。

本文首先给出几个基本概念和模型。随后,提出一个基于信任度的信任管理框架,并讨论其具体实现中的几个关键问题。最后总结本文工作并展望下一步的工作方向。

*)本文受国家自然科学基金(No. 60273034);863项目(No. 2001AA11310, No. 2002AA116010);江苏省自然科学基金和高技术项目(BG2001012,BK2002203,BK2002409)资助。徐 锋 博士生,研究方向:分布对象技术、系统安全、电子商务应用。吕 建 博士,教授,博导,研究方向:形式化方法、分布对象技术、构件技术。

2 概念和模型

2.1 几个基本概念

软件服务：软件服务是指具有自描述、自包含和模块化特征的软件实体，通过网络媒介向外发布，用于构造新的软件服务或应用系统。

软件服务协同：软件服务协同是指为了达成某个特定的应用目标，在软件服务实体之间建立联系、管理交互和协调活动的过程。

信任：D. Gambetta 给出了一个一般意义下的信任定义[9]，指出信任应具有三个基本特征：1) 主观性，不同的个体对同一事物的看法会受个体喜好等因素影响而不同；2) 可能性预期，信任的程度可表示为对事件发生概率的可能性估计；3) 内容相关，信任是对事物的某个方面（如完成某项任务的能力）而言的。根据此定义，我们给出了软件服务协同语境下的信任定义，即在一个软件服务协同系统中，信任是指一个协同 Agent 对其他软件实体是否能够正确地、非破坏性地进行某项（类）协作活动的主观可能性预期，预测的依据来源于此前该 Agent 所观察到（包括其他可信任第三方提供）的目标服务的行为，预测结果受该 Agent 对此项协作活动的重要程度评价（如关键协作活动、次要协作活动等等）影响。我们据此提出了一个基于经验的信任度计算模型。

经验：信任评估主体对客体的观察结果和第三方提供的该客体的观察结果定义为经验，前者称为直接经验，后者称为推荐经验（间接经验）。对应于客体活动的成败，经验简单地分为成功经验和失败经验，其计数定义为经验值。

信任管理：M. Blaze 创造了“信任管理”一词，其定义为，采用一种统一的方法描述和解释安全策略（security policy）、安全凭证（security credential）以及用于直接授权关键性安全操作的信任关系（trust relationship）[10]。信任管理系统的核心内容是，用于描述安全策略和安全凭证的安全策略描述语言以及用于对请求和安全策略进行一致证明验证的信任管理引擎。D. Povey 结合主观信任度思想给出了一个更一般意义的信任管理定义，即信任管理是信任意向（trusting intention）的获取、评估和实施[11]。

2.2 软件服务协同模型

Internet 环境下的软件服务逐渐从简单的功能封装向能够自主适应服务调用对象和网络应用环境的方向发展。软件服务具有更多的主动性和智能性，由多个软件服务构造而成的应用系统（软件服务）更多地表现为软件服务实体间主动的协作活动。为完成某项任务或实现某个功能，满足应用需求（包括功能、性能、安全等方面）的多个软件服务实体结成协作联盟，并分别担任相应的组织角色。协作联盟能够随应用环境、协同目标以及联盟成员的变化而动态调整，已完成特定任务或不可用的软件服务可能被联盟排除，而新的软件服务也可能应某种需求而被联盟吸收。一个较为抽象的软件协同模型如图1所示。

软件服务是一个相对独立的软件实体，通常由应用逻辑和协调逻辑两部分组成，前者描述软件服务的功能，后者描述与其他软件服务之间的协调。其中，软件服务的协调逻辑被抽象为一个相对独立的协调 Agent，代表软件服务处理所有协调相关事务，多个软件服务之间的协调实际上是各自 Agent 之间的协调。协调 Agent 在与其他软件服务结成联盟时，负责寻找、评价协作伙伴，并进行结盟前的协商。当联盟建立后，协调 Agent 用于完成软件服务间应用语义相关的协作活动。为了便于多个软件服务之间的协调工作，我们引入一类具有协调联盟内各软件服务之间活动能力的软件 Agent，称之为组织者。该软件 Agent 可以是应用系统的发起服务方提供，也可以通过联盟内多个协调 Agent 选举产生，视具体的应用环境和需求而定。

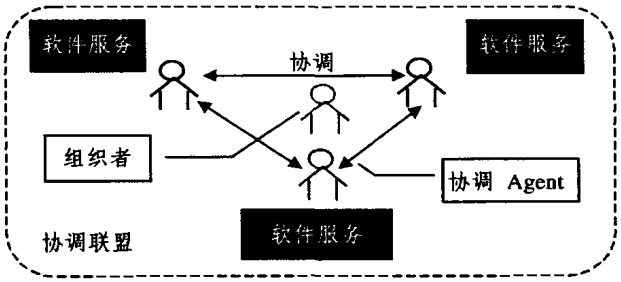

图1 软件服务协同模型

2.3 基于经验的信任度计算模型

根据上述信任定义，信任度计算模型的主要目的是通过软件服务实体所获得的经验信息对软件服务之间的信任关系进行度量。基于经验的信任度计算模型的主要内容包括：

1) **信任度与经验信息的关系** 假设软件实体 A 为信任评估主体，B 为信任评估客体。A 对 B 的信任关系评估可用 B 能以 A 期望的成功概率或可接受的失败概率完成某项（类）协作活动的可能性来度量，该度量的依据来源于 A 所获得的关于 B 的相关经验信息。

设 A 对 B 关于 ω 类协作活动的成功经验值和失败经验值分别为 m_ω 和 n_ω，以及 A 对 B 完成此类协作活动的成功率期望为 α。A 对 B 的信任度 V_ω 可由式(1)计算获得：

$$V_\omega=P_\omega(X\leqslant m_\omega)=\sum_{l=1}^{m_\omega}C_{m_\omega+n_\omega}^{l}\alpha^{l}(1-\alpha)^{m_\omega+n_\omega-l} \tag{1}$$

2) **经验信息的综合** 信任度评估主体的经验信息可以由可信任第三方推荐而获得，另一方面，评估主体对经验推荐者的信任程度不同，将导致其对推荐者所提供的经验信息的采纳程度不同，模型中使用采纳系数来表示，为0～1之间的实数。某个直接经验信息通常需要通过多个经验推荐者，最终传递到信任评估主体，从而形成了经验推荐路径。另外，信任评估主体往往需要从多个不同的经验推荐路径上获得经验信息，以便得出更合理的信任度判断。经验信息综合处理应能消除多路径推荐中产生的重复经验信息并计算出合理的经验值。

设 $E_i(i=1,\cdots,m)$ 分别为经验推荐路径上各不相同的最终推荐者，其对评估客体 B 的直接经验值为 M_i^d 和 N_i^d，$C_{i,j}(i=1,\cdots,m,j=1,\cdots,n_i)$ 为各推荐路径的等效采纳系数，其中 $C_{i,\cdot}$ 是一类以 E_i 为最终推荐实体的推荐路径的等效采纳系数，而 n_i 是该类推荐路径的个数。为方便描述，定义 C_i 为 E_i 到 A 的等效采纳系数，单条推荐路径的等效采纳系数 C_i 可由式(2)计算获得：

$$C_i=\begin{cases}1 & \text{if } (\forall\ j\in[1\cdots n_i])\wedge(\exists\ C_{i,j}=1)\\ 1-1/\sum_{j=1}^{n_i}\dfrac{1}{1-C_{i,j}} & \text{else}\end{cases} \tag{2}$$

设 A 对 B 的直接经验值为 M_d 和 N_d，可通过式(3)、(4)计算最终的一致的经验值 M，N：

$$M=M_d+\mathrm{int}\left(\sum_{i=1}^{m}C_i\cdot M_i^d\right) \quad (3)$$

$$N=N_d+\mathrm{int}\left(\sum_{i=1}^{m}C_i\cdot N_i^d\right) \quad (4)$$

3 基于信任度的信任管理框架

基于信任度的信任管理框架在传统的基于策略的信任管理框架的基础上加入了与信任度评估有关内容，如经验信息的收集机制、信任度评估机制等。其中信任度评估机制是基于经验的信任度计算模型具体实现。另外，扩充了安全策略描述语言的描述能力和使用范围，增加了对信任度的支持，并用于描述经验信息收集和信任度评估等相关的策略。其主要组成部分有：一致性验证机制、策略描述语言、本地策略数据库、安全凭证系统和信任度评估机制等部分组成，其中一致性验证机制和信任度评估机制是信任管理框架的核心部分，称之为信任管理引擎。各组成部分及其相互关系如图2所示。

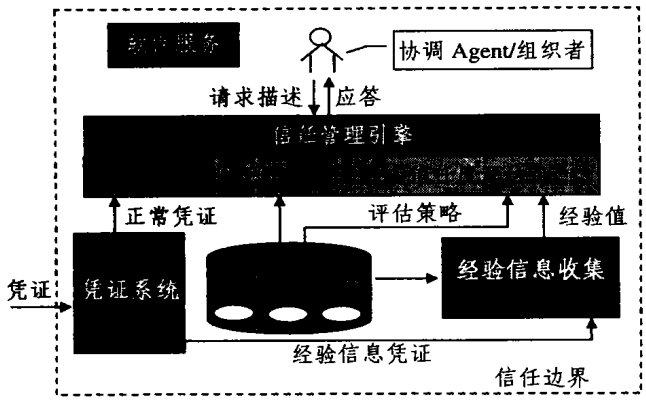

图2 信任管理框架

软件服务协调的安全问题可分为两个方面：1）软件服务个体之间的安全问题，其目标是保护各软件服务的私有资源，各服务实体对应的协调 Agent 通过信任管理机制进行点对点的信任管理，以决定是否进行协作活动；2）协调联盟的安全问题，其目标是保护联盟共有的服务资源，协调组织者处于信任管理的中心，通过设置特定的安全和信任策略，赋予不同的协调参与实体相应的资源访问权限。联盟组织者和软件服务协调 Agent 均可使用上述信任管理框架进行安全管理，其具体过程描述如下：

软件服务所有者使用安全与信任策略描述语言事先定义软件服务协调 Agent 或联盟组织者的安全策略、经验信息收集策略和信任度评估策略等，并将这些策略存放于信任管理系统的本地策略数据库中。软件服务协调 Agent 或联盟组织者输入其他软件实体发出的协调请求描述，经验信息收集机制收集该软件实体的相关经验信息和安全凭证，经验信息包含在一类特殊的经验信息安全凭证中。收集到的经验信息作为信任度评估机制的输入部分，按照信任度计算模型进行计算得出该软件实体的信任度，并与其他非经验信息安全凭证作为一致性验证机制的输入部分，一致性验证机制根据信任度、其他安全凭证等输入信息判断协调请求是否符合本地安全策略，并输出结果。软件服务协调 Agent 依据该结果决定是否与其他软件服务建立协作联盟，以及协作联盟建立后如何保障自身系统的安全，即在协作活动中依据对协作伙伴不同的信任程度选择适宜的安全措施对恶意行为进行甄别、防范和事后追查。而联盟组织者，则依据该结果决定联盟的组织结构，和各协调实体对共享资源的访问权限。

本地策略数据库是信任管理框架的关键部分，其他组成部分如一致性验证机制、经验信息收集和信任度评估机制的工作过程受存放其中的相关策略控制。所存放的策略按其功能可分为三类：

·授权和认证策略，用于委托或授权其他软件服务执行某类协作活动或访问本软件服务资源，包括定义对其他经验信息安全凭证签发者的信任程度，用于一致性验证过程；

·信任评估策略，用于推荐信任程度的划分、度量，以及指定信任信息的综合和计算过程中所使用的各类参数，用于信任度评估过程；

·经验信息收集策略，用于限定信任传递路径的长度，推荐信任采纳的阈值等，用于经验信息的收集过程。

基于信任度的信任管理框架对传统的安全凭证进行了扩充，增加了一类用于传递包含信任评估客体经验信息的安全凭证，经验推荐者使用策略描述语言和统一的格式描述信任评估客体的经验值及其来源等经验信息，此类安全凭证仅用于信任度评估。

4 系统实现的几个关键问题

实现上述信任管理框架主要应解决几个关键性问题，安全与信任策略描述语言的设计和实现，安全凭证的加密和签名和信任管理引擎等。

4.1 安全与信任策略描述语言

从便于安全策略和安全凭证的交流，以及易实现和易用性的角度考虑，我们选择使用 XML 语言来实现信任管理框架中的安全与信任策略描述语言。我们还拟实现一个辅助用户进行策略编辑的图形化输入界面，使用户能够较方便地完成安全和信任策略的设置工作。几个主要的 XML 标记(tag)说明如下：

〈POLICY〉标记：本地策略使用〈POLICY〉标记定义，具有两个属性：NAME，策略名；TYPE，策略类型，对应本地策略的功能分类，分别是 Security、Trust valuation 和 Experience Collection。一个〈POLICY〉标记的作用范围内可以包含一个或多个〈RULE〉标记，用于定义该策略的实际内容。

〈CREDENTIAL〉标记：安全凭证使用〈CREDENTIAL〉标记定义，具有三个属性：NAME，凭证名；TYPE，凭证类型，用于区分普通的非经验信息凭证和经验信息凭证，分别是 Normal 和 Experience；ISSUER，安全凭证的签发者。一个〈CREDENTIAL〉标记的作用范围内同样可以包含一个或多个〈RULE〉标记，用于定义该安全凭证的实际内容。

一个规则定义了一些实际的策略或安全凭证内容，最常见的内容有委托或授权其他软件实体验证某个协同请求或访问本地资源、经验信息收集策略和设置信任度评估策略等。为此，我们定义了多个标记以满足不同规则内容的描述需求，如〈FUNCTION〉、〈EXECUTE〉、〈COLLECTION〉等。其中〈FUNCTION〉标记用于支持简单的数值或字符串比较操作，当需要进行一些复杂的计算操作时，可以使用〈EXECUTE〉标记调用外部处理程序，该标记具有输入、输出等属性用于与外部处理程序交换数据，〈COLLECTION〉标记专门用于定义经验信息收集中推荐路径长度和广度的限定。一个规则可以包含多个非运算标记，考虑到信任管理引擎的实现，每个规则

（下转第161页）

网相连，供用户运行并行程序的节点要与系统网相连，控制台只与高速以太网和标准以太网相连。整个系统通过外部网连接到 Internet 上，这样，一般用户可通过 Internet 上机，系统管理员也可通过 Internet 登录到控制台上对系统进行管理和维护(也允许系统管理员登录到系统中任何一个节点上实施管理操作)。所有节点还通过串口线连接，当内部网失效时，可以用于故障诊断。系统中节点的标准配备是9GB 的硬盘，可以在控制台上外挂各种基本的外部设备，包括光驱、磁带机和打印机，系统也根据需要在系统中的一些服务节点上外挂各种外部设备，如 RAID。用户也可增加其它外设(如三维图形显示、磁带机、打印机等)外挂到机柜9、机柜10或控制台上。

结语 本文讨论了当前最流行的三种并行计算机系统 SMP、MPP 和 COW，COW 的性能优于 MPP，所以机群技术是今后可扩展并行计算机的主流发展趋势。

参 考 文 献

1 Hwang K. Scalable Parallel Computing: Technology Architecture Programming, McGraw Hill, 1998

2 Culler D E. Parallel Computer Arichitecture. A Hardware/Software Approach. Morgan Kaufmann, 1998

3 李国杰. 曙光一号并行计算机. 计算机学报，1994，17(11)：882～889

4 Agerwala T. SP2 system architecture. IBM Systems Journal, 1995, 34(2): 154～184

(上接第154页)

只能最多包含一个运算标记，如〈EXECUTE〉、〈FUNCTION〉。

下面给出使用上述策略语言描述的一个指定经验收集过程中推荐路径长度、广度以及采纳满足某个推荐信任度等内容的本地策略，其中推荐路径的搜索深度为3，搜索广度为4，并且只有经验推荐者的推荐信任度大于0.8，其推荐经验信息才被采纳。该策略的具体描述如图3所示。

```
〈POLICY NAME="selling experience collection" TYPE="Experience Collection"〉
    〈PULE〉
        〈COLLECTION ID="selling" LENGTH="3" WIDTH="5"〉〈/COLLECTION〉
        〈FUNCTION〉
            〈GT〉
                〈FIELD ID="Recommendation Reliability"〉〈/FIELD〉
                〈CONST〉0.8〈/CONST〉
            〈/GT〉
        〈/FUNCTION〉
    〈/RUL〉
〈/POLICY〉
```

图3 本地策略描述实例

4.2 安全凭证的加密和签名

信任管理系统中的安全凭证是软件实体之间传递信任信息的载体。为了能够有效地保护安全凭证的内容在传递过程中不被篡改、伪造、泄露以及抵赖，必须在安全凭证传递之前对其进行加密和数字签名，并在凭证接收之后对其解密和签名验证。

信任管理系统中，一个安全凭证实际上表现为一个符合策略描述语言语法的 XML 文档。从经济和易实现的角度考虑，我们在安全凭证的加密和数字签名实现中遵行了安全 XML 规范[12]，并使用了一些第三方提供的符合安全 XML 规范的软件开发包。

4.3 信任管理引擎

信任管理引擎是信任管理框架的核心部分，不同于传统的信任管理系统，我们增加了信任度评估模块，其具体实现符合我们提出的基于经验的信任度计算模型。一致性验证模块的实现参考了几个已有的信任管理系统[4~6]，但我们的实现较为简单，将在以后的工作中完善。

结论 我们分析了当前软件服务协调环境下安全问题呈现的新特点和需求，指出了传统的基于策略的信任管理系统在解决上述软件环境安全问题中存在的不足，并在此基础上提出了一个结合信任度评估的信任管理框架，较好地弥补了传统信任管理系统的不足。此外，我们初步实现了一个信任管理原型系统，验证了其可行性，并体现了该信任管理框架的易操作性、合理性和策略设置的灵活性。

需要指出的是，信任管理框架的实现较为初步，有待进一步地增强策略描述语言的描述能力和一致性验证模块的证明验证能力。相信完整的系统实现，将更能体现该信任管理框架在解决软件服务协同安全问题中的优越性。

参 考 文 献

1 Herrmann P, Krumm H. Trust-adapted enforcement of security policies in distributed component-structured applications. In: Proc. of the 6th IEEE Symposium on Computers and Communications. Tunesia: IEEE Computer Society Press, 2001. 2～8

2 Khare P, Rifkin A. Trust Mangement on World Wide Web. World Wide Web Journal, 1997, 2(3): 77～112

3 Bhargava B, Zhong Y. Authorization Based on Evidence and Trust: [CERIAS Tech Report]. 2002

4 Blaze M, Feigenbaum J, Lacy J. Decentralized Trust Management. In: Proc. 17th Symposium on Security and Privacy. Oakland: IEEE, 1996. 164～173

5 Blaze M, Feigenbaum J, Ioannidis J, Keromytis A D. The KeyNote Trust Management System Version 2. Internet RFC 2704, Sep. 1999

6 Chu Y-H, Feigenbaum J, LaMacchia B, Resnick P, Strauss M. REFEREE: Trust Management for Web Applications. World Wide Web Journal, 1997, 2(2): 127～139.

7 Beth T, Borcherding M, Klein B. Valuation of Trust in Open Network. In: Proc. European Symposium On Research in Security (ESORICS). Brighton: Springer-Verlag, 1994. 3～18

8 JØsang A. A model for trust in security systems. In: Proc. of the Second Nordic Workshop on Secure Computer Systems. 1997

9 Gambetta D. Can We Trust Trust?. In: Trust: Making and Breaking Cooperative Relations. Basil Blackwell, Oxford, 1990. 213～237

10 Blaze M, Feigenbaum J, Ioannidis J, Keromytis A. The Role of Trust Management in Distributed Systems Security. In: Secure Internet Programming: Issues for mobile and distributed objects. Berlin: Springer-Verlag, 1999. 185～210

11 Povey D. Developing Electronic Trust Policies Using a Risk Management Model. In: Proc. of the 1999 CQRE Congress. Nov. 1999. 1～16

12 Ford W, Hallam-Baker P, Fox B, et al. XML Key Management Specification(XKMS). W3C Note. 2001