

SOA 和 Web Service 的区别

董洛兵, 尚博, 李玉斌

(西安电子科技大学 图书馆, 陕西 西安 710071)

摘要: SOA(Service Oriented Architecture) 技术无疑为下一代企业分布式架构的趋势, 但是很多人目前还无法准确把握 SOA 的概念。本文旨在帮助读者认清 SOA 与 Web Service (Web 服务) 的本质区别。文章从分布式计算的发展历程出发, 说明 Web Service 与 SOA 的出现背景及本质区别。

关键词: 分布式架构; SOA; Web Service; 分布计算; RPC; ORB; MOM

中图分类号: TP302 文献标识码: A 文章编号: 1009- 3044(2008)10- 20040- 03

The Difference between SOA and Web Service

DONG Luo- bing, SHANG Bo, LI Yu- bin

(Xidian University Library, Xi'an 710071, China)

Abstract: This paper helps people to distinguish between SOA and Web Services Based on analyzing the history of distributed computing, it describes SOA and Web services, and proceeds to outline the differences between them.

Key words: distributed computing; SOA; Web Service; distributed architecture; RPC; ORB; MOM

1 前言

SOA(Service Oriented Architecture), 即以服务为导向的软件开发思想, 是当前技术界最热门的话题, 目前, 关于 SOA 本质及实现方法的文章很多, 而且用 Web Service(Web 服务) 技术作为核心构件去搭建 SOA 架构逐渐成为主流, 但是多数文章都混淆 SOA 与 Web Service 的概念, 没有明确指出二者的本质区别, 导致很多人误认为二者本质上是相同的, 或只是简单的扩展关系。

本文旨在帮助读者认清 SOA 与 Web Service 的本质区别。要想理解一种技术的本质, 从了解其发展历史出发无疑是最简单的办法, 本文从分布计算的发展历程出发, 引出 Web Service 出现的原因, 从而说明 Web Service 的本质, 然后透析 Web Service 与 SOA 的本质区别。

本文分五部分, 前言部分说明问题的来源。第二部分说明分布式技术的发展历程中几个重要技术的特点; 第三部分说明 Web Service 技术的本质; 第四部分说明 SOA 架构出现的背景、实质以及与 Web Service 的区别; 第五部分总结与展望。

2 分布计算的发展历程

通信中间件的出现无疑是分布式计算技术发展史上的一次重大变革, 在早期的分布计算中, 两个分布式程序之间的通信在原始物理网络协议的基础上直接实现。编程人员必须处理物理网络的准确细节。通信中间件的出现, 封装了低级通信机制的技术复杂性, 对应用程序开发人员隐藏通信技术库的细节。

卓越的中间件框架提高了分布式应用程序的灵活性、互操作性、移植性和可维护性, 成为有效实现分布式软件架构的关键。下面按出现的先后顺序说明三种典型的通信中间件的特点:

2.1 RPC(Remote Procedure Calls, 远程过程调用)

RPC 将 '本地过程调用' 的概念应用到分布式应用程序。RPC 使得对远程函数的调用就如同对本地函数的调用一样(如图 1)。

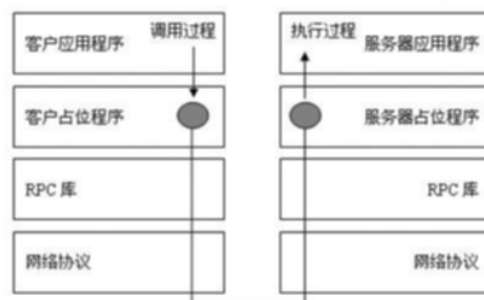


图 1 RPC 通信原理

大多数 RPC 实现都基于同步式请求—应答协议。

收稿日期: 2008- 03- 02

作者简介: 董洛兵(1981-), 男, 西安电子科技大学图书馆, 助理工程师, 研究生; 尚博(1981-), 男, 西安电子科技大学图书馆, 助理馆员, 研究生; 李玉斌(1967-), 男, 西安电子科技大学图书馆, 工程师, 研究生。

2.2 分布式对象

分布式对象技术建立在 RPC 之上, 由 ORB(Object Request Broker, 对象请求代理) 支持, ORB 管理与可能的远程对象的通信和数据交换。ORB 基于“互操作对象引用”概念, 允许通过对象工厂和其他辅助对象, 方便地远程创建、定位、调用和删除对象, 如图 2 所示。

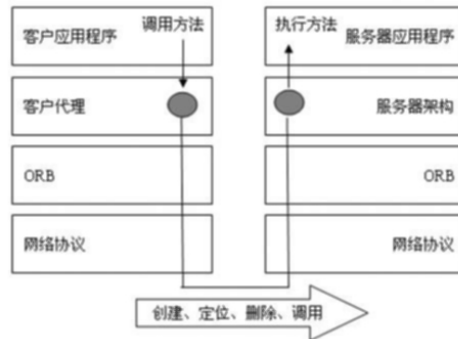


图 2 ORB 通信原理

ORB 技术提供了一个面向对象的分布式平台, 促进了跨计算机、软件和供应商的对象通信。当前主要 ORB 实现有 CORBA、DCOM、RMI, 虽然这些 RPC 中间件技术已广泛应用于各个领域, 但是面对规模和复杂度都越来越高的分布式系统, 这些技术也显示出其局限性: (1) 同步通信: 客户发出调用后, 必须等待服务对象完成处理并返回结果后才能继续执行; (2) 客户和服务对象的生命周期紧密耦合: 客户进程和服务对象进程都必须正常运行; 如果由于服务对象崩溃或者网络故障导致客户的请求不可达, 客户会接收到异常; (3) 点对点通信: 客户的一次调用只发送给某个单独的目标对象。

2.3 MOM(Message Oriented Middleware, 面向消息的中间件)

MOM 较好的解决了以上问题, 常见的 MOM 都是基于消息队列来实现的: 发送者将消息发送给消息服务器, 消息服务器将消息存放在若干队列中, 在合适的时候再将消息转发给接收者。如图 3 所示。

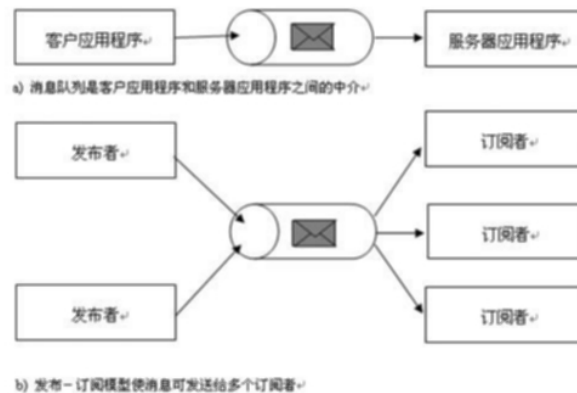


图 3 MOM 通信原理

已有的 MOM 系统包括 IBM 的 MQSeries、Microsoft 的 MSMQ 和 BEA 的 MessageQ 等。由于没有一个通用的标准, 这些系统很难实现互操作和无缝链接。

3 Web Service 的产生

虽然中间件技术为单个系统的开发提供了卓越的基础结构, 但由于分布式计算概念、标准和产品过多, 产生了“中间件异质性”问题。大多数中间件的初衷是解决“应用程序异质性”问题, 却引发了令人尴尬的“中间件异质性”问题, 事实证明, 根本无法在大型企业中强制执行企业范围的中间件标准。XML 技术的出现很好地解决了这个问题, XML 是一种流行的、独立于中间件的格式, 可以在不同应用程序之间交换数据和文档。XML 基本上是 IT 行业可以达成一致的最小公分母。XML 不与特定技术或中间件标准绑定, 经常作为一种特殊格式来处理各个不兼容中间件平台的数据。1998 年, 微软工程师利用 SOAP(Simple Object Access Protocol, 简单对象访问协议) 发明了基于 XML 的 Web Service(Web 服务)。

Web Service 本质上是要求以标准化的方式实现企业内外各个不同服务系统之间的互调或者集成。这里从一种基本的方法来分析一下这个目标的实现方法。假如服务 A 要远程调用服务 B 上的服务, 要实现这个目的, 需要下面 3 个要素:

(1) 服务 B 要以一种标准化的语言告诉服务 A 它能提供什么样的服务, 以及如何调用它的服务, 它的服务在哪里等, 这就是 Web Service 的服务描述, 是 What、How 和 Where 部分。

(2) 服务 A 要以一种标准化的通信消息格式告诉服务 B, 它想用什么服务, 并加入相应的输入参数等。当服务 B 完成服务后, 会同样以标准化的通信方式告诉 B 相应的服务结果, 是 Web Service 的服务消息的 request 和 response 部分。

(3) 服务 B 要注册到相应的公共网址, 以便外部能找到, 是 URL 部分。

以上 3 个要素实际上对应于 Web Service 的 3 个组成部分——WSDL、SOAP 和 UDDI。如图 4 所示:



图4 Web Service 的三个组成部分

WSDL (Web Service Description Language) 主要目的在于 Web Service 的提供者将自己的 Web 服务的所有相关内容, 如所提供的服务的传输方式、服务方法接口、接口参数、服务路径等, 生成相应的完全文档, 发布给使用者。SOAP 是 Web Service 的标准通信协议, 是一种标准化 XML 格式的传输消息, 以便大家都用同一种格式来讲话, 可以相互完全理解, UDDI (Universal Description、Discovery and Integration) 是一种创建注册服务的规范, 以便大家将自己的 Web Service 进行注册发布供使用者查找。

Web Service 技术实现了服务接口的传输和调用的标准化, 服务接口和服务实现的分离, 以及 Web 服务组件的可重用性。

4 SOA 技术和 Web Service 的本质区别

4.1 SOA 技术的产生

虽然 Web Service 技术实现了服务接口的传输和调用的标准化, 但是还是没有解决企业分布技术的异质性, 反倒使这种异质性更加扩大, 如图 5 所示:

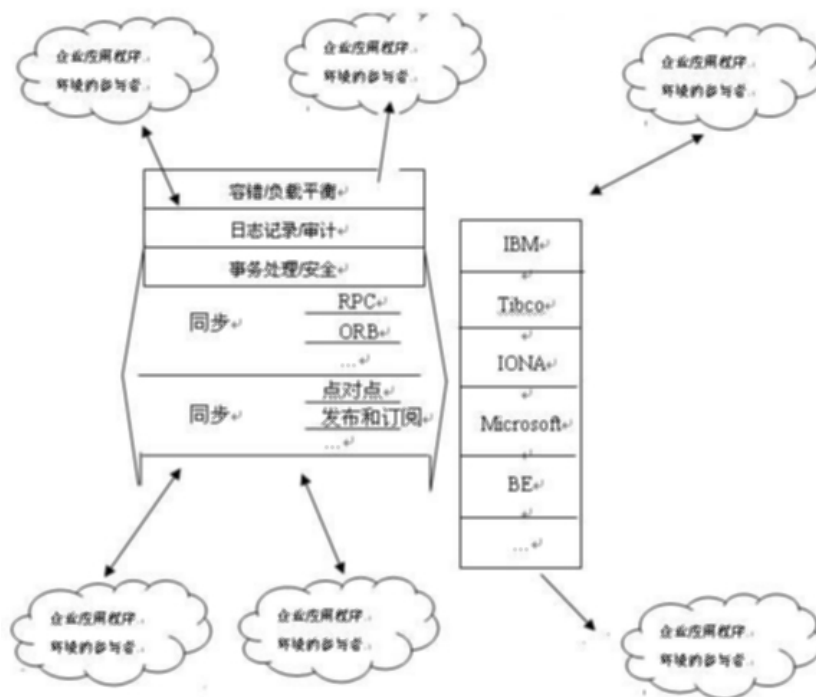


图5 企业分布技术的异质性

异质性(特别是中间件的异质性)的存在是一个基本事实, 我们无法阻止它, 却可有效地管理它。所以要求现代架构必须支持所有这些技术和概念, 同时必须适应底层分布基础结构的不断变化, 正是这些需求推动了 SOA 技术的产生。

4.2 SOA 的实质

SOA 是一个组件模型。它将应用程序的不同功能单元(成为服务)通过这些服务之间定义良好的接口和契约联系起来。图 6 显示了 SOA 架构的示意图。

SOA 整合发布平台将完全无关的平台所提供的各种服务整合起来一起发布给外界, 包括实施安全控制和监控服务状态等, 客户端完全不知道真正的服务发布者是谁。

4.3 SOA 和 Web Service 的根本区别

SOA 和 Web Service 都包含“服务”的概念, 并且都能对客户端封装服务的具体实现, 所以很多人简单的认为 SOA 和 Web Service 本质上是相同的, 或者 SOA 只不过是 Web Service 的一种应用而已。但是两者是有本质的差别的, Web Service 只是实现 SOA 的一种主要途径而已。

(下转第 47 页)

第二类攻击是对部署了网络入侵防御系统的特定网段发动的攻击。这种攻击类型被认为是网络入侵防护系统最难解决的问题之一。当系统受到攻击而出现多形态的缓冲区溢出现象时,由于大量的状态管理和处理的需要,网络入侵防御系统被迫选择是让恶意流量通过还是在处理攻击流量的同时面临耗尽资源的可能性,最终可能导致那个网段的拒绝攻击服务。

3.3.3 信任指数

在入侵防护模型中,保证较低的误报率是至关重要的。对于传统的入侵检测系统来说误报是最令人讨厌的,在嵌入式防护模型中对攻击的错误判断可能导致合法流量的终止。把这种情况应用于一个公司的网络,一次丢失的交易可能导致几百万美元的损失。

考虑到这些潜在风险,是否收到一个报警通知就要防护主要取决于对这个报警的准确性的信心。通过衡量对已检测到的攻击的信息来决定是否触发防护就是我们所称的“信任指数”。信任指数是衡量报警信息可信度的重要措施。这是根据一些参数计算的。信任指数涉及的变量参数包括报警类型、明确使用的标识符数量、周期性的流量状态,以及经用户确认的准确性记录等^[7]。

根据信任指数,网络入侵防护系统可以设置成不同的报警类型,当达到某个特定的信任指数时才触发防护。例如,安全管理员可以决定只防护那些信任指数达到 90 或更高的攻击。这允许用户根据自己对风险的不同承受能力来衡量和设置防护触发的起点。

4 结束语

企业网络的安全架构需要不断发展来迎合未来的需求,对入侵管理的传统观念也需要相应的转变。入侵管理技术的新发展不仅要能满足企业规模的需要,还必须满足企业管理的需要,降低设备安装、升级和管理的复杂性,利用智能检测机制将误报率减至最低。在这种新的安全模型下,仅仅通过检测-报警的机制对攻击做出响应是远远不够的,以可信、智能的方式防护攻击将是入侵管理系统的主要功能。IIMS代表了新一代智能入侵检测和防护技术,它针对现有产品的不足做出了很有意义的改进。IIMS通过智能入侵检测、高级管理和可信防护三个方面,为下一代的入侵管理产品奠定了基础。

参考文献:

- [1] 周建国,曹庆国,赵庆军. 计算机网络入侵检测系统的研究[J]. 计算机工程, 2003,29(2):176- 178.
- [2] 蒋建春,马恒太,任党恩,等. 网络安全入侵检测研究综述[J]. 软件学报, 2000,11(11):1460- 1466.
- [3] 刘春颂,杨寿保,杜滨. 基于网络的入侵检测系统及其实现[J]. 计算机应, 2003,23(2):152- 153.
- [4] 吕汇新. 一个基于模式匹配入侵检测技术的防信息泄露系统的设计与实现[J]. 哈尔滨师范大学自然科学学报, 2004,20(3):52- 55.
- [5] 胡华平,等. 面向大规模网络的入侵检测与预警系统研究[J]. 国防科技大学学报, 2003,25(1):82- 83.
- [6] 刘昕,吴秋峰,袁萌. DDOS(分布式拒绝服务)研究与探讨[J]. 计算机工程与应用, 2000,36(5):131- 133.
- [7] 唐正军. 入侵检测技术导论[M]. 北京:机械工业出版社, 2004,185- 186.

(上接第 42 页)

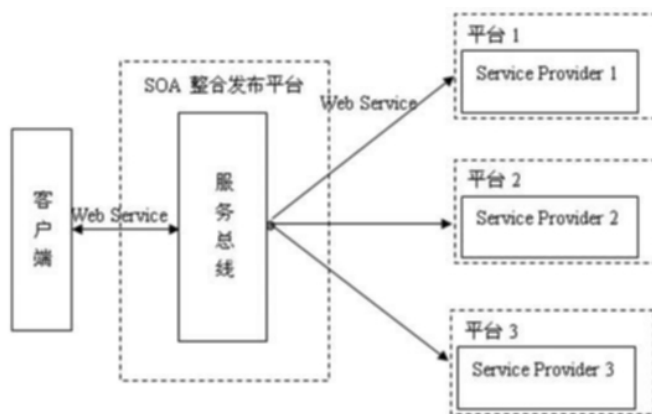


图 6 SOA 架构

Web Service 服务接口需要绑定具体实现的服务组件来实现服务,它对具体的服务实现完成了封装,但是它本身是知道服务是如何实现的,另外客户端调用 Web Service 组件时,需要知道 Web Service 的具体位置和传输协议。但是 SOA 架构平台只和服务接口进行绑定,实现了服务接口的透明化,服务位置的透明化,服务传输协议的透明化。SOA 本身也不知道服务具体是如何实现的。SOA 实现了更高层次的抽象。

5 总结与展望

本文通过介绍通信中间件技术的发展过程,引出 Web Service 的本质,从而进一步说明 Web Service 和 SOA 的本质区别,能够为读者正确把握 SOA 架构的概念提供很好的帮助。

SOA 技术顺应了现有企业应用软件架构的发展趋势,必将成为下一代企业软件架构的主力。

参考文献:

- [1] 梁爱虎.精通 SOA[M].电子工业出版社,2007.
- [2] 韩宏志,译.Enterprise SOA 中文版[M].清华大学出版社,2006.