学校代码：0357
学　　号：

安徽大学

硕士学位论文

论文题目(中文)

Web 环境下数据仓库安全技术研究

The Research of Security Technology of data Warehouse Under the Web Environment

论文题目(英文)

姓　　名　　袁学松
学科专业　　计算机应用技术
研究方向　　网络和数据库
指导教师　　谢荣传

完成时间　　2005年5月

摘 要

数据仓库的安全性是一个非常重要的问题。不同的安全技术被用来解决这个问题。本文首先对数据仓库的安全研究进行了回顾。针对 WEB 环境下的数据仓库中存在的安全隐患，从数据仓库的建设和运行两大方面入手，从技术性和非技术性的角度出发，提出了相应的解决措施。通过对 XML 的各种安全技术的详细讨论，在论文中提出了一个使用 XML 及相关安全组件实现的 WEB 环境下的数据仓库安全模型。该模型中综合运用了 XML 加密、XML 签名、SOAP、XML 防火墙等技术来保证数据仓库的安全，为研究 WEB 环境下的数据仓库的安全提供了一种新的思路。论文的最后还给出了通过 XML Security Suite 实现 XML 签名的例子。

关键词：WEB、数据仓库、安全、XML、模型

Abstract

The security of a data warehouse is the most critical issue. Different techniques are used to deal with it. In this thesis we first review the strengths and weakness of the latest techniques and practices .We discussed the possible hole in the data warehouse under the WEB circumstance and proposed the corresponding technique and non-technique solution. On the basis of the discussion of the XML and related security technologies, we proposed a security model using XML and its related technologies to ensure a better and secure data warehousing experiences. This model combined some technologies of XML encryption, XML signature, SOAP, XML firewall which is presented a new idea for the security of a data warehouse under the WEB circumstance. An example of XML signature using XML Security Suite attached in the end of the paper.

Key words: WEB、Data warehouse、security、XML、model

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得安徽大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名：袁学松　　签字日期：2005 年 5 月 8 日

学位论文版权使用授权书

本学位论文作者完全了解安徽大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权安徽大学可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名：袁学松　　导师签名：

签字日期：2005 年 5 月 8 日　　签字日期：2005 年 5 月 8 日

学位论文作者毕业去向：

工作单位：三联职业技术学院信息系　　电话：0551-3830796

通讯地址：合安路47号三联职业技术学院　　邮编：230601

第一章 绪论

1.1 论文的研究背景

1.1.1 数据仓库技术的应用和发展

数据仓库[1]是近年来兴起的一种新的数据库应用。它的出现和发展是计算机应用到一定阶段的必然产物。经过多年的计算机应用和市场积累，许多商业企业已经保存了大量的原始数据和各种业务数据，这些数据真实地反映了商业企业主体和各种业务环境的经济动态。然而由于缺乏集中存储和管理，这些数据不能为本企业进行有效的统计、分析和评估提供帮助。也就是说，无法将这些数据转化为企业有用的信息。数据仓库正是顺应这种要求而产生的。随着各种计算机技术如数据模型、数据库技术和应用开发技术的不断进步，数据仓库技术也在不断发展，并在实际应用中发挥了巨大的作用。因特网资料中心在 1996 年对 20 世纪 90 年代前期进行的 62 个数据仓库项目的调查研究结果表明：进行数据仓库项目开发的公司在平均 2.73 年的时间内获得了平均为 321%的投资回报率。使用数据仓库所产生的巨大效益的同时又刺激了数据仓库技术的需求，数据仓库市场正以迅猛势头向前发展：一方面，数据仓库市场需求量越来越大，每年约以 400%的速度扩张；另一方面，数据仓库产品越来越成熟，生产数据仓库工具的厂家也越来越多。据 IDC 数据，2000 年全球数据仓库市场为 128 亿美元，较上年增长 20%，到 2003 年，全球数据仓库市场为 250 亿美元，年增长率为 25%[2]。

进入 20 世纪 90 年代，特别是 1995 年以来，WEB 技术得到了迅速发展。作为新一代软件技术，WEB 能够在不同的网络及操作环境中运行，并可以方便地扩展到外部的相关企业及最终客户，这很好地适应和促进了互联网应用的发展。与此相适应，基于 WEB 的应用由于继承了 Internet 的固有优点而受到了广泛的欢迎。WEB 技术的飞速发展对数据仓库产生了很大的影响。数据仓库提供了自由使用存储信息的途径，而利用 WEB 技术可以方便、经济地获得有关信息。这两种技术的结合，就产生了使信息分布和处理更经济、更高效的基于 WEB 的数据仓库系统。基于 WEB 的数据仓库技术改变了最终用户对数据仓库的使用模式。人们可以不再局限于通过局域网（LAN）使用数据仓库，而是可以通过 Internet/Intranet 远程访问数据

仓库，所得的分析结果也可以借助 WEB 服务器迅速发布。对于扩大数据仓库的使用范围，提高信息的使用效率具有较大的意义。

1.1.2 网络信息安全问题的日益严重

信息化和网络化是当今世界经济和社会发展的大趋势，也是推进我国国民经济和社会现代化的关键环节。计算机技术和网络技术已深入到社会的各个领域，人类社会各种活动对计算机网络的依赖程度已经越来越大。与此同时，由于计算机网络所具有的开放性和共享性，其安全性也成为人们日益关切的问题。在世界范围内，对计算机网络的攻击手段层出不穷，网络犯罪日趋严重，给各行各业带来了巨大的经济和其他方面的损失。据美国律师联合会安全调查，有40%的被调查者承认在他们的机构中曾经发生过计算机犯罪的事件。报道的黑客入侵事件在1990年为252起，1994年增至2341起，2004年全球一共发生了392,545起。据美国FBI估计，计算机网络每被攻破一次造成的损失为50万美元，而一个大银行的数据中心停机一秒的损失为5000美元。2000年伪造支票造成的金融服务系统的损失为27亿美元，信用卡伪造所造成的损失也达到35亿美元。这些数字所显示的仅仅是美国网络安全犯罪所造成的真正损失的一小部分。许多机构还未意识到在他们的机构中存在计算机犯罪。此外，即使发现了犯罪事件，许多机构也不愿意公开它们的存在。据专家估计，由计算机犯罪所造成的实际的经济损失每年高达150亿美元。很多国家机构和企业等网络用户的网络安全措施做得都不理想，我国也属于网络安全性意识薄弱的国家。据国家公安部消息，国内有超过80%的网站缺乏安全措施，因此造成了严重的网络安全隐患。比如由于2001年中美撞机事件所引起的中美黑客大战，我国至少有350个网站的主页被改的面目全非。而近年爆发的计算机病毒，更是严重给我国的银行等金融网络系统造成了巨大损失。

1.1.3 数据仓库安全分析

随着数据处理功能的不断细化以及 Internet 互联作用的日益渗透，数据仓库的运用日渐普及和深入，其数据蕴涵的价值正被广泛认可和重视，相应的安全问题也备受关注。影响数据仓库系统安全的原因主要有以下两个方面：

（1）数据价值提升[5]。数据仓库系统中的数据从数据源提取出来，历经多种处理过程，数据价值已经提升。存储在数据仓库系统中的数据大致分为概要数据和

细节数据[6]：前者从宏观上展示事物的发展规律，对于决策制订等重要分析活动起着关键作用，数据概括度越高，其价值越大；而后者详细反映了事物变化的具体过程，往往具有很强的敏感性。

（2）数据共享的形式、范围以及程度正出现新的变化[6,7]。Internet/Intranet是使这个过程变得异常开放的重要因素；另一方面，全球化的历史步伐使企业的结盟愈演愈烈，数据共享已经从拥有者个体内部逐渐走向个体之间，甚至出现跨行业的多重交叉共享。

这两个原因直接威胁着数据仓库系统的安全。

在基于Web的数据仓库系统中，由于数据是大量集中存放，而且用户对数据仓库的应用不再局限于单一的环境，它可以为互联网上众多用户直接共享。数据仓库的分析结果往往要求能够在网上发布、浏览，让用户在线使用。这种方式在给企业的应用带来便捷的同时，也为数据仓库带来了安全隐患。概括来说，主要有以下几种类型的非法访问：

(1)恶意破坏。这类非法访问是一种人为的破坏，访问数据仓库的主体往往不是合法的用户，这种行为的目的就是要给数据仓库中的数据造成破坏。如黑客，这些人不具备访问数据仓库的合法的权限，但通过黑客技术入侵系统，恶意访问数据。

(2)非法入侵。这类用户访问数据仓库的目的不是破坏，而是期望从数据仓库中获得有价值的信息。但这些信息对他们来讲是禁止访问的。非法入侵的用户往往是合法的用户，但他们滥用权限蓄意窃取信息。由于他们处在局域网内部，对网络的情况比较了解，有一定的网络使用权限，如果没有相应的安全措施，他们的行为非常容易实现。例如，一个用户对某个已创建的报表有浏览的权限，但如不加以控制．他可以利用已有的权限对报表的钻取访问创建基于报表的数据库表；还有一种被称为特洛伊木马的程序，这种程序具有很强的隐蔽性，能利用用户的合法权限对数据的安全进行攻击。

(3)非法误用。有些用户虽然拥有合法的权限、主观上不存在恶意的访问，但在操作过程中也可能因为误操作给数据仓库中的数据造成破坏和泄漏。

由于我国的数据仓库应用还处在发展的初级阶段，近年来虽然一些数据仓库产品开始关注安全问题，增加了一些安全措施，但还不够规范。基于Web的数据仓库

的安全对策就更少。但我们应该清醒地意识到，随着网络技术和数据仓库技术不断紧密的结合以及应用发展的需求，必须尽快地给出基于Web数据仓库的安全性策略。这也就是本文将要讨论和解决的问题。

1.2 论文的研究意义

基于 WEB 的数据仓库系统的创建是一项既具有挑战性又有益的工作。与传统的数据仓库相比，基于 WEB 的数据仓库具有易于访问、操作平台无关和管理成本低等明显优势[3]，为企业提供了更有力的的决策支持，大大提高了企业的经济效益。随着基于 WEB 数据仓库技术应用的推广，越来越多的企业依赖它进行各种重要的业务分析和决策制定，这些都是维系企业良好发展的高层应用。没有这些高层应用企业就不能确信下一步应该如何走好；同样，缺少数据仓库技术的支撑，也就不能建立有效的高层应用。这种相互依存的特性使得数据仓库系统正逐渐演变成拥有者的致胜工具和无形资产，其重要性不言而喻。如果数据仓库系统存在安全隐患，如：数据泄密、数据损坏等，这往往会给企业带来巨大的伤害，甚至波及企业的存亡，更为严重的是可能给国家的经济带来严重的损失。正是由于数据仓库的重要性，研究数据仓库的安全便显得尤为重要。这也就是本论文将要讨论和解决的问题。通常，数据仓库系统存储着企业大量的历史数据和重要企业逻辑，然而，由于数据仓库连接到网络上，理论上任何浏览器用户都可能通过网络(Intranet /Extranet /Internet)来访问。这使得基于 WEB 的数据仓库中数据的安全受到威胁，从而影响企业的决策。基于 WEB 的数据仓库的安全研究的意义在于根据现有的数据仓库系统存在的安全隐患进行分析，找出影响安全的因素，并因此制定相应的安全策略，采用一定的安全技术，最大限度地保证数据仓库系统的安全，保证重要数据和企业逻辑的安全，减少企业因为缺乏安全而带来的损失，使得数据仓库系统能够更好地为企业决策提供服务。

1.3 主要的研究内容

本课题是 WEB 环境下的数据仓库安全技术的研究，研究的主要内容包括：

1、对数据库和网络信息系统的安全进行研究

数据仓库的安全是一个非常复杂的问题，它涉及到多个方面。由于数据仓库和数据库之间存在着直接的联系，现阶段对数据仓库安全的解决大多还是从传统

数据库技术出发，所以本文中对与数据仓库安全密切相关的数据库的安全问题进行了研究；在 WEB 环境下的数据仓库中，由于信息基于网络进行传输，因此网络的安全问题也是影响数据仓库安全的重要因素。本文对网络信息系统的安全问题也进行了研究。

2、对 XML 及相关安全技术进行研究

WEB 环境下的数据仓库能够把政府或企业中的分散的原始操作数据和各种来自外部的数据汇集成一个单一的数据集合，并且通过 WEB 传送系统提供给各种不同用户。这时，如何将来自不同应用系统的分散的数据集合汇集成一个比较合理的数据集就成为比较迫切的问题。以 XML 作为中间层的数据描述工具和数据转换工具可以提供一种比较良好的解决方案。将 XML 应用于数据仓库具有很多优势。正是因为 XML 的优越性，越来越多的公司通过网络用 XML 来传输结构化的数据，文档的安全问题也越来越重要。本文对 XML 及相关的安全技术如 XML 加密、XML 签名等进行了研究。

3、对 WEB 环境下的数据仓库安全技术进行研究

在前面研究的基础上，对 WEB 环境下数据仓库中存在的安全隐患进行了分析，从数据仓库的建设和运行两方面入手，从技术性和非技术性的角度出发，对 WEB 环境下的数据仓库的安全进行了研究。

1.4 论文的成果和组织

1.4.1 研究成果

本论文在前人研究的基础上，结合国内外的最新发展动态，通过大量的资料采集、整理和分析，对 WEB 环境下的数据仓库的安全进行了分析和研究。

本文所取得的成果主要有：

（1）对现有的数据仓库系统存在的安全问题和主要的安全措施（网络的安全防护措施、操作系统的安全防护措施、数据库的安全防护措施）进行了分析和探讨，研究了基于 WEB 的数据仓库中存在的安全隐患并提出了相应的解决措施。

（2）基于对 XML 及相关安全技术的研究，提出了一个用 XML 安全组件构建的基于 WEB 的数据仓库安全模型。给出了该模型的组成部分，模型的具体工作步骤和涉及到的主要技术问题，还给出了模型中 XML 签名实现的例子。

1.4.2 论文的组织

本文安排如下：

第一章： 绪论

介绍了选题的研究背景和意义，论文主要的研究内容和要解决的问题。

第二章： 数据仓库安全研究回顾

通过国内外文献的研究，对当前数据仓库系统安全的相关技术予以全面的回顾、比较和总结。

第三章： 数据库和信息系统的安全

分别介绍了数据库和网络信息系统的安全问题。在数据库安全里，介绍了数据库的安全评估标准，分析了数据库中存在的安全问题，给出了保证数据库安全的主要手段；在网络信息系统安全里介绍了网络信息安全的基本要求和存在的安全威胁，网络安全体系结构及网络安全的相关技术。

第四章： XML及相关安全技术

介绍了XML及相关的安全技术，如XML签名、XML加密、XACL、XKMS、SOAP、XML防火墙等内容，为后面介绍基于XML的安全模型打下基础。

第五章： WEB环境下数据仓库的安全研究

针对WEB环境下的数据仓库中存在的安全隐患，从数据仓库的建设和运行两大方面入手，从技术性和非技术性的角度出发，研究提出了相应的解决措施。

第六章： 基于XML的数据仓库安全模型

提出了一个基于XML及相关安全技术的WEB环境下数据仓库的安全模型。给出了模型的具体工作步骤和模型中部分实现的例子。

第七章： 结论和展望

总结全文，给出结论并提出了进一步的研究方向。

第二章 数据仓库安全研究回顾

2.1 传统环境下数据仓库安全研究回顾

下面将按照不同的技术路线，回顾数据仓库安全的研究，并做出适当的解释和比较，最后予以总结。

2.1.1 工程实践

如果能够在实施数据仓库的过程中就能够充分考虑到数据仓库的安全性，势必能够在系统的安全措施的实施方面占据主动。基于这种思想，Warigon 在文献[8]中把数据仓库的安全需求划分了七个阶段，它们分别是：

（1）数据鉴别－这是以下阶段的关键，建议收集、整理出一套关于数据仓库中数据的详细清单，内容涉及到数据的内容、性质、位置以及最终用户的使用情况等。

（2）数据分类－依照安全保密性的要求对数据进行分类，可以根据数据的敏感程度和重要性分为公开数据、机密数据和高度机密数据。

（3）数据价值的定量化－数据在使用过程中可能会遭受破坏、泄密等各种安全问题，因而要对这些事件发生的可能性以及恢复过程付出的代价进行估算，最终反映为量化的数据价值，以便重点保护关键数据。

（4）确定数据仓库系统的安全隐患－主要包括：

- DBMS 的内置安全（基于视图的安全可以被直接的数据卸载方法旁路）
- DBMS 的限制（访问高密级数据的应用程序泄漏低密级数据）
- 双重安全引擎（将 DBMS 和操作系统的安全措施进行联合实施，不仅会导致安全问题的偏离，也会增加管理和维护工作的复杂性，不便于问题的诊断）
- 推论攻击（利用数据之间存在的各种联系，通过低密级数据推算出高密级数据）
- 可用性因素的限制（安全要求同数据仓库系统的访问共享理念相悖）
- 人为因素（过失、疏忽、误用、破坏、妨碍、欺骗等意外行为及主动行为）

- 内部威胁（内部员工破坏数据的保密性）
- 外部威胁（竞争对手和其他外部团体刺探企业决策数据）
- 自然因素（火、水等自然灾害）
- 公共设施因素（电力、通讯服务的中断）

（5）防护措施及实现代价的确定－防护措施包括：对操作人员的安全培训、访问用户的分类、访问控制、完整性控制、数据加密、隔离设置、开发质量控制等。

（6）选择性价比高的安全措施－权衡上一阶段中安全措施的功能以及实现代价，选择、搭配合适的安全措施予以实施。

（7）评估安全措施的功效－对安全措施实际运用的效果进行评价，并按照实际功效不断调整和优化，进一步提高整个系统的安全性。

Warigon 从工程实践的角度进行了周全的、阶段性的安全考虑，对实施安全的数据仓库工程具有指导意义。

2.1.2 安全体系结构

由于大的数据仓库系统往往涉及到多个异构的数据库系统，这和联合数据库系统有着很多相似之处。通过借鉴联合数据库系统的研究成果，Abelló 等人将其中先进的系统构造模式运用于数据仓库的概念设计环节中，其主要集成过程是[9]：

（1）按照数据驱动的原则选定包含所需分析数据的联合模式（Federated Schema）。

（2）将联合模式改造成数据仓库模式（DW Schema），主要是增加时间元素以体现数据的历史性，可采用数据仓库中常用的星型模式（Star Schema）等。

（3）根据实际使用的安全策略（比如：基于 MAC 的多级安全）定义用户/应用程序能够访问到的数据集合，构造出授权数据仓库模式（Authorization DW Schema），用来描述数据仓库逻辑模式的安全访问策略。

（4）定义外部数据仓库和数据集市模式（External DW and Data Mart Schema），以便进行数据分析。

上述各个模式的定义采用 BLOOM 数据模型语法，整个构造过程如图 2.1-所示。

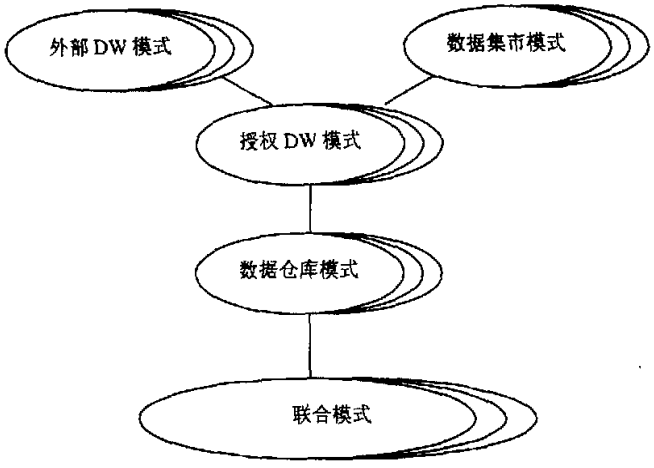


图 2.1 数据仓库安全模式体系结构

联合数据库的分层模式本身是一个比较完备、层次分明的理论架构，有助于从整体上把握数据仓库系统的安全设计。

2.1.3 数据加密

数据加密是一个非常古老的话题，它通过改变数据本来面貌使得未被授权用户无法知道数据的本来面目来防止泄密。鉴于数据仓库系统中数据的重要性，文献[7]和[10]分别就数据的存储和传播提出了中肯的建议和意见：数据加密存储属于内部安全措施，它可以用来防止直接对数据的物理形式进行卸载(dump)。而数据的加密传播适合于分布式数据仓库应用环境，可以防止数据在网络上的泄密问题。

加密是一把双刃剑，隐藏数据的直观表现是它理想的一面，但为此需要付出相当大的代价：

（1）最明显的事实是加密/解密过程增加了处理器的处理时间[8]。

（2）大多数数据仓库系统为提高数据的访问效能，可根据数据内容进行寻址，而加密后的数据会破坏这个条件。

（3）为了增强加密的效果，在数据传播中采用动态加密技术，这对客户端来说也是一个不小的负担。

数据仓库中的数据加密不同于传统的消息加密[6]，存在较多的加密限制，具体包括：加密数据不能进行比较、搜索操作，所有出现外键的字段必须统一加密，加密数据不能进行计算，主键不能被加密等[7]。

由于加密确定可以提高系统的安全性，比较切实可行的数据仓库加密原则是：

最好不进行大规模的加密，而是选择安全性要求较高的数据进行局部加密。也就是只加密那些敏感的、关键的、而又不过多损耗系统性能的数据。

2.1.4 访问控制

访问控制在数据库系统中的研究相当深入，但由于数据仓库和数据库的语义存在很大差别，加之访问控制是一个信息系统最基本的安全措施，所以大部分安全研究都集中在数据仓库的访问控制方面。以下介绍的是一些学者提出的几个基于访问控制的数据仓库的安全模型：

（1）基于角色的数据仓库安全模型

角色是数据库中一个很重要的概念。通过角色的定义可以加强数据库的安全性。由于数据仓库中的数据存储与数据库中的数据存储有着诸多的不同，所以将数据库中角色的实现照搬到数据仓库中是行不通的。

Remzi Kirkgoze,Nevena Katic 于 1997 年提出一种基于角色的 OLAP 数据立方体访问控制策略[5]。这个模型是一个基于 AMAC（Adapted Mandatory Access Control）的控制 OLAP 多维立方体的安全控制策略。在这个模型中主要描述了数据仓库中每个角色的安全限制规则。每个用户有一个角色，每个角色对应一个安全规则表。这些规则表组成了角色的安全限制文件。根据这个角色安全限制文件，每个用户可以访问到他被授权访问的数据。

这个模型是一个只有“角色”作为安全主体（Security Subject）的数据仓库安全模型，该模型权限只能赋予角色。安全对象（Security Object）是安全系统中的客体，在 OLAP 环境中维表、事实表及它们的属性都是安全对象。每一个安全主体被授权能对安全对象进行某种操作，这些操作称为访问类型。

该模型的特点是能为不同的子立方体赋予不同角色，并能为系统用户赋予不同角色，达到灵活性与安全性的统一。

（2）基于授权的安全模型

授权是指一个对象对系统对象或者系统本身所拥有的合法访问权限。Edgar Weippl 等人提出了一种对数据仓库和 OLAP 的一个授权模型[11]，基于数据仓库中多维数据的存在，他们通过一种简单的描述符号来描述了这种访问授权，具有比用 SQL 的授权机制更直观的特点。

该模型可以直观表达数据仓库中的多维数据模型的基本元素，包括主体对象、

访问类型、客体对象和谓词。其中客体对象指的是数据仓库中多维立方体中的维度、维度中的层次、事实表。访问类型主要考虑六种基本的操作：Read、drill-Down、Roll-Up、Slice、Dice and Drill-Through。

考虑数据仓库中的一个销售链例子，包括四个维度：（1）时间（全部、年、月、日）（2）地理位置（全部、省份、地区、县市、销售点）（3）商品（全部、产品、子类别、类别）（4）销售度量（单价、利润）。允许用户“王丽”访问每种产品的单价，但是不允许访问每种产品的利润，允许访问子类别或者类别。同时，她也不能检索每年或者每天的合成数据，但是允许访问每个城市或商店的数据。可以通过下面的语句来实现：

(王丽，{单价},{(时间，{(read,全部)，(Drill-Down,月)})，(地理位置，{(Drill-Down,省份),(Roll-Up,地区)})，(商品，{(Drill-Down,类别),(Roll-Up,产品)})})。

该模型主要是根据数据库中 SQL 的语言机制演变而来，根据数据仓库中多维数据的特点，提出通过一种更适合数据仓库中多维数据访问的策略。用一套简单的描述符号对于任意一种给定的安全策略，都可以很容易的实现它的访问控制权限。

（3）基于元数据的数据仓库安全模型

元数据是关于数据的数据，是数据仓库中的数据字典，主要描述数据仓库内数据的结构和构建方法的数据。它是数据仓库中很重要的一部分，影响数据仓库中所有的层次，常被开发者用来管理控制和开发数据仓库。没有元数据用户不能正常访问数据仓库。基于这种思想 N.Katic 等人提出了一个基于元数据的安全模型[12]。

文献[13]中把数据仓库中的元数据分为结构元数据和访问元数据两类。其中结构元数据主要是指用来创建和维护数据仓库，它主要描述了数据仓库的结构和内容。访问元数据描述了数据仓库和终端用户之间的动态关系。

N.Katic 等人在其 WWW-DIS-DWH 项目中根据用户的访问需要，为不同的用户组构造不同的访问元数据，这些元数据都是来自总的元数据文件。新加入的用户加入一个用户组，就具有了这个组的访问权限。通过在数据仓库上添加一个“安全管理者”（管理、定义、描述用户和用户群体）和一个“安全查询管理层”（作用

是通过检查是否允许一个任务的执行来过滤用户的查询）来实现的。

此模型主要是通过“安全管理者”和“信息服务器”可以把用户想查询的而没有查询权限的那部分数据给过滤掉，只返回给用户他可以访问的那些数据，此操作对于用户来说是透明的。因为用户不知道自己有部分数据被过滤掉了，从而减少了去试图访问他原本看不到的数据的可能性，这就增强了数据仓库中数据的安全性。

（4）基于视图的数据仓库安全模型

数据仓库中的数据来源于多个数据源，每个数据源系统对数据都有不同的安全性需求，这些数据的安全性需求也需要反映到数据仓库中。从这个角度，A. Rosenthal 等人提出了一种基于视图的数据仓库安全模型[14]。基本思想是：把数据源和数据仓库看作同一分布式数据库的一部分，这样在数据仓库中就可以沿用数据库中的视图访问机制。就是说允许自动配置对数据仓库的许多访问控制。从某种意义上说，简化了管理过程。原理可用图 2.2 示意。

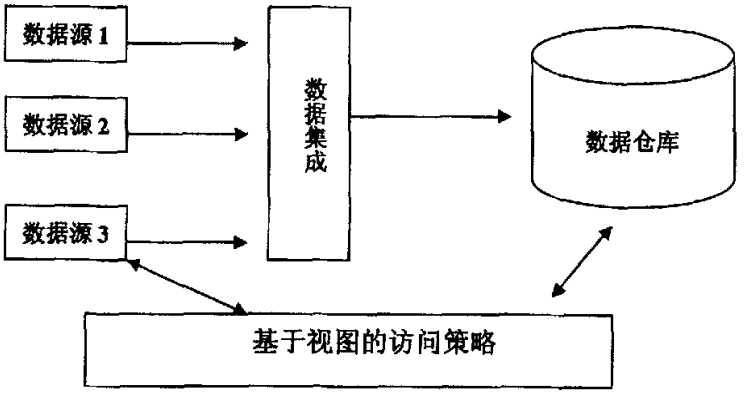


图 2.2　基于视图的数据仓库安全模型示意图

基于视图的这种安全模型的特点是充分考虑了数据库中视图机制，利用数据库中的视图机制使得数据仓库的安全策略易于管理。系统能够自动的从数据源分析用户对于数据的访问权限，然后根据这种访问权限自动的生成用户在数据仓库中能够访问哪些数据。这样就减少了技术人员在数据仓库中进行相应方位策略的配置问题，同时能够很好的解决数据源发生变化之后的问题。

2.1.5　审计

前面介绍的安全技术路线都是以主动的形式进行的，而审计则是在事情发生之

后开展的一项安全活动，所以被称为被动的方法。审计是对访问控制的必要补充，是访问控制的一个重要内容。审计会对用户使用何种信息资源、使用的时间、以及如何使用（执行何种操作）进行记录与监控。审计和监控是实现系统安全的最后一道防线，处于系统的最高层。审计与监控能够再现原有的进程和问题，这对于责任追查和数据恢复非常有必要。

审计跟踪是系统活动的流水记录。该记录按事件从始至终的途径，顺序检查、审查和检验每个事件的环境及活动。审计跟踪通过书面方式提供应负责任人员的活动证据以支持访问控制职能的实现（职能是指记录系统活动并可以跟踪到对这些活动应负责任人员的能力）。审计跟踪记录系统活动和用户活动。系统活动包括操作系统和应用程序进程的活动；用户活动包括用户在操作系统中和应用程序中的活动。通过借助适当的工具和规程，审计跟踪可以发现违反安全策略的活动、影响运行效率的问题以及程序中的错误。审计跟踪不但有助于帮助系统管理员确保系统及其资源免遭非法授权用户的侵害，同时还能提供对数据恢复的帮助。

审计技术在操作系统、数据库等传统领域中早已运用。但由于数据仓库的组织形式和访问特点，这方面的进展并不明显。由于数据仓库系统往往缺乏自己的管理系统，通常要依赖已有的数据库管理系统，而这些系统的审计只能反映 SQL 的语义，很难体现数据仓库的访问特点，只有转换成数据仓库的语义，才能够使审计对数据仓库的安全有所帮助。

数据仓库中使用审计的关键所在是把审计安放在系统架构中的合适位置[16]。文献[15]中主要描述了在审计数据仓库时需要注意的一些问题，对数据仓库的审计分成三个部分进行：抽取数据、存储数据、访问数据。如图 2.3 所示。

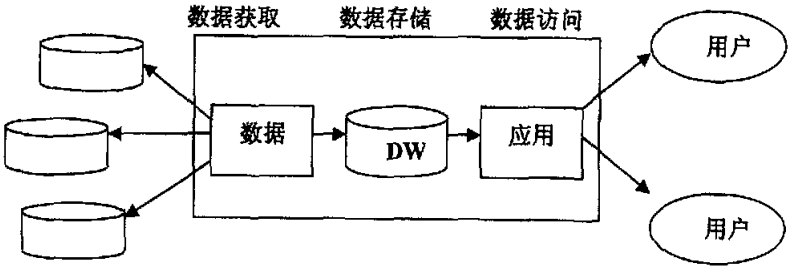


图 2.3 数据仓库及其环境

然后分别讨论了这三个部分在审计的过程中应该注意的一些问题。尤其强调

了数据访问过程的审计，同时说明系统的操作也应该受到监视，尤其是查询和表格最为敏感。

2.1.6 OLAP 安全

OLAP 是数据仓库技术最典型的应用，Priebe 等人在 GOAL 项目中专门针对 OLAP 应用提出了安全设计方法学[16]。它由若干独立的阶段组成。如图 2.4 所示。方法的主要特点是将安全定义和实现相互分离，利于相关技术的变化、调整和适应。

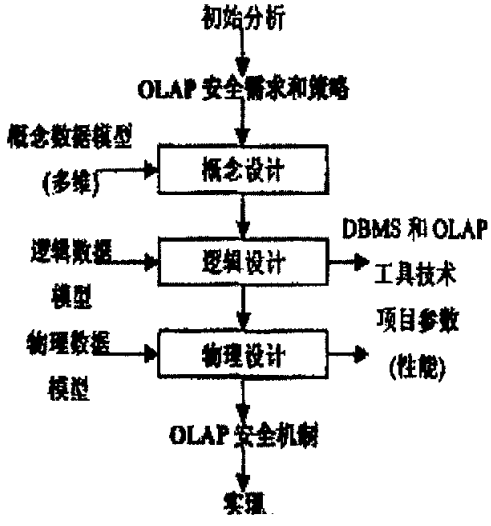


图 2.4 OLAP 安全设计方法学

因为 OLAP 的许多操作依赖于大量的概要数据，推论控制显得格外麻烦。虽然信息推论早在统计数据库安全中就有研究，但由于该问题的复杂性，到目前为止，还缺乏切实可行的解决办法。推论控制是这些安全需求中最难解决的问题。除此之外，该文献还对常见的 OLAP 商业产品进行了安全需求的横向比较和评估，这些信息对于设计实际系统的安全措施、选择 OLAP 产品具有很好的指导作用。

同时，作者从信息隐藏的角度，按照多维数据的概念提出了十项安全要求，如表 1 所示。

作为信息隐藏思想研究的后继工作，Priebe 正式提出了多维安全约束语言（Multidimensional Security Constraint Language）-MDSCL[17]，该语言主要针对“角色”隐藏多维数据，初步实现了表 1 中的基本需求。

表 1 不同的 OLAP 访问控制需求

2.1.7 商业产品研究

目前，数据仓库的商业产品有很多种，如 IBM、Oracle、Sybase、CA、NCR、Informix、Microsoft 和 SAS 等有实力的公司相继（通过收购或研发的途径）推出了自己的数据仓库解决方案，BO、Brio 和 Cognos 等专业软件公司也在前端在线分析处理工具市场上占有一席之地。这些数据仓库商业产品对于 OLAP 安全访问控制方面也有些不同。文献[16]中讨论了部分数据仓库商业产品对多维数据 OLAP 的安全访问控制的实现情况。如表 2 所示。

几种商业产品对于数据仓库中多维数据 OLAP 信息的隐藏也不尽相同。上述几种产品都实现了文献[16]中对 OLAP 进行隐藏的基本需求，即隐藏整个立方体、隐藏某些度量、隐藏立方体的某些切片以及隐藏细节层次。此外对于 Cognos PowerPlay6.0 还可以隐藏同一维度中某些切片的细节层次。而 Oracle 中只有最高两层的推论控制和动态/数据驱动策略没有实现。

表 2 数据仓库商业产品的安全特性评估表

产品 / 信息	基于 ROLAP 的产品	Microsoft SQLServer2000	MicroStrategy	Cognos PowerPlay	Oracle Express
评估版本	N/A	8.0BETA	7.0BETA	6.0	6.2
安全实现方法	SQL 视图	Cell 层和维度安全	访问控制列表和安全过滤	用户类和维视图	Permission 程序
实施安全的架构部分	DBMS	OLAP 服务器和 OLAP 前端	OLAP 服务器和 OLAP 前端	OLAP 前端	OLAP 服务器
一般方法	视图	混合	视图	视图	规则
实施规则	封闭式	开放式	开放式	开放式	开放式
安全管理员	数据所有者	管理员	数据所有者	管理员	管理员

2.1.8 小结

上面分别从工程实践、安全体系结构、数据加密、访问控制、审计、OLAP 安全及商业产品七个方面对传统环境下的数据仓库安全研究现状进行了回顾，就技术路线而言，对数据仓库安全的解决大多还是从传统数据库技术出发，其主要原因在于数据仓库同数据库之间的直接联系，由此可以得出这样的结论：数据仓库建立的目的是如何组织好数据便于访问，因而对于安全性的研究比较少；而另一方面，从实际数据仓库运行的环境来看，又迫切需要安全防护，所以我们能够做的就是从数据存储、数据访问、数据传播等传统角度来扩展数据仓库的访问语义，增加数据仓库系统的安全性。

由于数据仓库系统的存在形式大多借助于传统系统来支撑，而这里最直接的安全措施就是访问控制，原有系统的访问控制对于数据仓库的支持存在明显不足之处，访问控制就成为目前数据仓库系统安全防护的重心。

2.2 WEB环境下的数据仓库安全研究回顾

WEB环境下数据仓库传统的安全问题仍然存在。目前还没有多少专门研究WEB环境下的数据仓库安全的论文，唐蕾、徐洁磐等提出了一个基于WEB的数据仓库安全策略模型及标准[18]，重点讨论了在Web环境下的数据分析结果怎样安全共享的一

些策略与机制。Bill Inmon在其《Data warehouse and internet security》一文中提出，在WEB环境下传统的安全技术如logon/logoff、防火墙、基于应用程序的安全、基于视图的安全措施等仍然起作用，提供安全的高层次的方法就是对数据进行加密[7]，他还对数据仓库中的加密问题进行了分析。

总的来说，国内对于数据仓库安全的研究还处于起步阶段，这和数据仓库的应用情况及对该问题的认识程度有关，国外学者的研究思想和所取得的成果值得我们借鉴，希望本课题可以为数据仓库安全的研究带来新的活力。

第三章 数据库与网络信息系统的安全

随着计算机技术的飞速发展和企业界不断提出新的要求，数据仓库技术应运而生。传统的数据库技术是单一的数据资源，即数据库为中心，进行从事事务处理、批处理到决策分析等各种类型的数据处理工作。近年来，随着计算机的应用，网络计算开始向两个不同的方向拓展，一是广度计算，一是深度计算，广度计算的含义是把计算机的应用范围尽量扩大，同时实现广泛的数据交流，互联网技术的发展和普及就是广度计算的特征；另一方面就是人们对以往计算机的简单数据操作，提出了更高的要求，希望计算机能够更多的参与数据分析与决策的制定等领域。特别是数据库处理可以大致地划分为两大类：操作型处理和分析型处理（或信息型处理）。这种分离，划清了数据处理的分析型环境与操作型环境之间的界限，从而由原来的以单一数据库为中心的数据环境发展为一种新环境：体系化环境。

数据库系统作为数据管理手段，从它的诞生开始，就主要用于事务处理。经过数十年的发展，在这些数据库中已经保存了大量的日常业务数据。传统的业务系统一般是直接建立在这种事务处理环境上的。随着技术的进步，人们试图让计算机担任更多的工作，而数据库技术也一直力图使自己能胜任从事务处理、批处理到分析处理的各种类型的信息处理任务。后来人们逐渐认识到，在目前的计算机处理能力上，根本无法实现这种功能，而且，另一方面，事务处理和分析处理具有极不相同的性质，直接使用事务处理环境来支持决策是行不通的。要提高分析和决策的效率和有效性，分析型处理及其数据必须与操作型处理及其数据相分离。必须把分析型数据从事务处理环境中提取出来，按照 DSS 处理的需要进行重新组织，建立单独的分析处理环境，数据仓库正是为了构建这种新的分析处理环境而出现的一种数据存储和组织技术。

数据仓库是数据库系统发展到一定阶段的一种必然要求，可以说数据仓库是在数据库发展的基础上产生的，和数据库有着密不可分的联系，从某种意义上讲，数据仓库可以称为大的数据库，只是按照不同的主题和技术来组织数据。正是由于它们之间的联系，因此研究数据仓库的安全问题就必须要考虑数据库的安全问题。WEB技术的飞速发展，扩展了数据仓库的应用范围，使得数据仓库的访问变得更加方便。由于信息基于网络进行传输，就不得不考虑网络信息系统的安全问题。

下面分别对数据库安全和网络信息安全加以介绍。

3.1 数据库安全概述

数据库系统的安全性要求同其他计算机系统的安全性要求有很大的相似性。数据库安全是指保护数据库数据不被非法访问和非法更新，并防止数据的泄漏和丢失。

数据库系统，一般可以理解成两部分：一部分是数据库，按一定的方式存取数据；另一部分是数据库管理系统（DBMS），为用户及应用程序提供数据访问，并具有对数据库进行管理、维护等多种功能。

数据库系统安全，包含以下两层含义：

第一层是指系统运行安全，它包括：法律、政策的保护，如用户是否有合法权利，政策是否允许等；物理控制安全，如机房加锁等；硬件运行安全；操作系统安全，如数据文件是否保护等；灾害、故障恢复；死锁的避免和解除；电磁信息泄漏防止。

第二层是指系统信息安全，它包括：用户口令字鉴别；用户存取权限控制；数据存取权限、方式控制；审计跟踪；数据加密。

3.1.1 数据库安全评估标准

随着人们对安全问题认识和对安全产品的要求不断提高，在计算机安全技术方面逐步建立一套评估标准，以规范和指导安全信息系统的建立、安全产品的生产，并能较准确地评测产品的安全性能指标。在当前各国制定和采用的标准中，最重要的是1985年美国国防部颁布的“可信计算机系统评估标准（TCSEC）”桔皮书（又简称为DoD85）。1991年，美国国家计算机安全中心（NCSC）又颁布了”可信计算机评估标准关于可信数据库管理系统的解释（TDI）”。我国也于1994年2月发布“中华人民共和国计算机信息系统安全保护条例”。在TCSEC中将安全系统分为四大类七个等级，其基本特征如表1所示：

TDI是TCSEC在数据库管理系统方面的扩充和解释，并从安全策略、责任、保证和文档四个方面进一步描述了每级的安全标准。按照TCSEC标准，D类的产品是基本没有安全保护措施的产品，C类产品只提供了初级安全保护措施，一般不称为安全产品。B类以上的产品是实行强制存取控制的产品，也是真正意义上的安全产

品。所谓安全产品均是指安全级别在B1级以上的产品。

表 1 可信计算机系统评估标准（TCSEC）

类	等级	定　义	基本特征
D		最小保护	基本无安全保护
C	C1	自主安全保护	初级自主存取控制、审计功能
	C2	受控存取保护	细化自主存取控制，实施审计与资源分离
B	B1	带标记的安全保护	基于标识的强制存取控制、审计功能
	B2	结构化保护	形式化安全模型表型，对所有主体与客体实施
			DAC 与 MAC，隐通道约束
	B3	安全域保护	安全内核，更强的审计功能，系统恢复功能
A	A	可验证保护	提供 B3 级保护并提供形式化验证

3.1.2 数据库面临的安全威胁

数据库面临着严重的安全威胁，具体情况可用图 3.1 来说明。

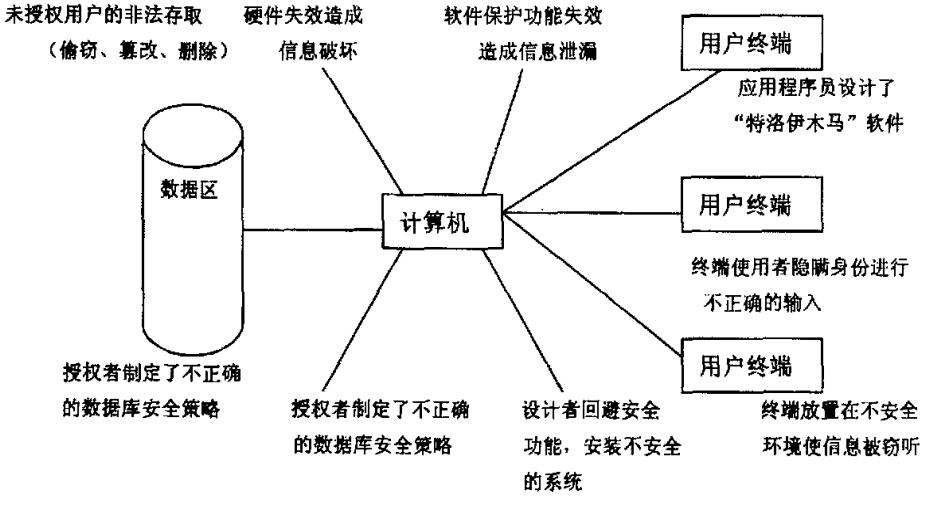


图 3.1 对数据库安全的威胁

对数据库安全的威胁主要来自以下因素：

（1）数据输入或处理中的错误，如准备输入的数据在输入前已被修改，有的机密数据在输入到计算机之前已被公开，在数据处理操作中的误操作等，均会使数据出错。

（2）硬件故障引起的信息破坏或丢失，如软盘或硬盘故障造成存储的信息丢失或破坏。

（3）软件保护功能失效造成信息泄漏，如操作系统设计上的缺陷，缺少存取控制机制或破坏了存取控制机制，造成信息泄漏。

（4）非授权用户的非法存取，篡改数据。如数据库管理人员对数据的使用权限不进行严格的管理、对哪些人有数据访问权、哪些人有数据修改更新权，心中无数，缺乏严格的检查控制措施；对用户在计算机上的活动没有进行监督检查。致使非授权用户非法存取，合法用户对数据进行篡改。

（5）授权者制定不正确、不安全的防护策略。

（6）操作者复制和泄露机密、敏感数据资料。

（7）系统设计者回避安全功能，安装不安全的系统。

（8）应用程序员设计、安装了”特洛伊木马”软件。

（9）终端放置在不安全的环境而被窃听。

（10）终端使用者隐瞒自己身份，进行不正确的输入。

（11）病毒侵入系统，破坏或修改了数据库软件。

数据库面对着各方面的严重威胁，要保证其安全、可靠，必须采用一定的安全策略和一定的安全技术措施，才能保证数据库中的信息不泄漏、不破坏、不被删除或修改。

3.1.3 数据库系统的安全性要求

（1）数据库完整性，它是数据库管理系统(DBMS)、操作系统(OS)和计算机管理者三方面应负的责任。数据库管理程序必须进行访问控制，确保只有授权用户才能进行数据更新或删除，另外还须防范非人为的外力灾难。从操作系统和计算机管理者看，数据库和 DBMS 分别是文件和程序，要保护数据库的完整性，必须周期性地对数据库文件进行备份．以预防由于灾难造成的损失。数据库的完整性包括物理上和逻辑上两种完整性。

（2）元素的完整性，它是指数据库元素的正确性和准确性，DBMS 要能帮助用户发现输入时的错误，并在输入错误数据后能及时纠正它们。DBMS 用三种方式维护数据库中每个数据元素的完整性。

①字段检查：这种检查可防止输入数据时可能出现的错误。

②访问控制：通过访问控制来保护数据库的完整性、真实性和一致性。

③更改日志：更改日志是数据库每次改变的记录文件，它包括文件记录原来的值和修改后的值，数据库管理员可以根据日志随时撤消任何错误的和非法的修改。

（3）可审计性．在某些应用中，可能需要产生对数据库的所有访问的审计记录，以帮助在事后发现发生过什么事，何人参加，有何影响，以协助维护数据库的完整性。数据库的审计踪迹包括对记录、字段和数据元素一级的访问。

（4）访问控制，DBMS 必须批准哪些数据可以访问，哪些数据禁止访问。其数据可以是字段、也可以是记录，或者是某个数据元。DBMS 可批准一个用户有权读、改变或删除或附加一个值，或者增加或删除整个字段或记录，或者重新组织数据库。

（5）用户认证，DBMS 应严格进行用户身份识别和认证。DBMS 可能要求用户输入口令和时间日期，以作检查。

（6）可获用性，数据库中的数据并不是任何时候部可以访问的。例如一个用户在更新几个字段时，其他的用户对这几个字段的访问请求便被禁止。当更新完毕时，其他用户对这些字段的访问即可获。

3.1.4 数据库系统基本安全架构

数据库系统信息安全性依赖于两个层次：一层是数据库管理系统本身提供的用户名、口令字识别、视图、使用权限控制、审计等管理措施，大型数据库管理系统 Oracle、Sybase、Informix 等均有此功能；另一层就是靠应用程序设置的控制管理,如使用较普遍的 Foxbase、Foxpro 等。对此，目前一些大型数据库管理系统(如 Oracle、Sybase 等产品)提供了以下几种主要手段。

1、用户分类

不同类型的用户授予不同的数据管理权限。一般将权限分为三类:数据库登录权限类、资源管理权限类和数据库管理员权限类。有了数据库登录权限的用户才能进入数据库管理系统,使用数据库管理系统所提供的各类工具和实用程序。同时,数据库客体的主人可以授予用户以数据查询、建立视图等权限。这类用户只能查阅部分数据库信息,不能改动数据库中的任何数据。具有资源管理权限的用户,除了拥有上一类的用户权限外,还有创建数据库表、索引等数据库客体的权限,可以在权限允许的范围内修改、查询数据库,还能将自己拥有的权限授予其他用户,可以申请审计。具有数据库管理员权限的用户将具有数据库管理的一切权限,包括访问任何用户的任何数据,授予(或回收)用户的各种权限,创建各种数据库客体,完成数据库的整库备份、装入重组以及进行全系统的审计等工作。这类用户的工作是谨慎而带

全局性的工作,只有极少数用户属于这种类型。

2、数据分类

同一类权限的用户,对数据库中数据管理和使用的范围又可能是不同的。为此,DBMS 提供了将数据分类的功能,即建立视图。管理员把某用户可查询的数据逻辑上归并起来,简称一个或多个视图,并赋予名称,再把该视图的查询权限授予该用户(也可以授予多个用户)。这种数据分类可以进行得很细,其最小粒度是数据库二维表中一个交叉的元素。

3、审计功能

审计功能是 DBMS 达到 C2 级以上安全级别必不可少的指标。众所周知,任何系统的安全保护措施都不是完美无缺的。提供审计功能的目的就是把任何人对数据库所作的任何操作记录在审计数据库中,DBA 通过阅读审计数据库,可以发现非法访问数据库的人、时间、地点以及所有访问数据库的对象和所执行的动作。大型 DBMS 提供的审计功能是一个十分重要的安全措施,它用来监视各用户对数据库施加的动作。有两种方式的审计,即用户审计和系统审计。(1)用户审计:DBMS 的审计系统记下所有对自己表或视图进行访问的企图(包括成功的和不成功的)及每次操作的用户名、时间、操作代码等信息。这些信息一般都被记录在数据字典(系统表)之中,利用这些信息用户可以进行审计分析。(2)系统审计:由系统管理员进行,其审计内容主要是系统一级命令以及数据库客体的使用情况。

4、数据库加密

一般而言,数据库系统提供的上述基本安全技术能够满足一般的数据库应用,但对于一些重要部门或敏感领域的应用,仅靠上述这些措施是难以完全保证数据的安全性,某些用户尤其是一些内部用户仍可能非法获取用户名、口令字,或利用其他方法越权使用数据库,甚至可以直接打开数据库文件来窃取或篡改信息。因此,有必要对数据库中存储的重要数据进行加密处理,以实现数据存储的安全保护。

1）数据库加密的特点

较之传统的数据加密技术,数据库密码系统有其自身的要求和特点。传统的加密以报文为单位,加密脱密都是从头至尾顺序进行。数据库数据的使用方法决定了它不可能以整个数据库文件为单位进行加密。当符合检索条件的记录被检索出来后,就必须对该记录迅速脱密。然而该记录是数据库文件中随机的一段,无法从中间

开始脱密,除非从头到尾进行一次脱密,然后再去查找相应的这个记录,显然这是不合适的。必须解决随机地从数据库文件中某一段数据开始脱密的问题。

2）数据库密码系统采用公开密钥

传统的密码系统中,密钥是秘密的,知道的人越少越好。一旦获取了密钥和密码体制就能攻破密码,解开密文。同时由于数据库中的数据是共享的,有权限的用户随时需要知道密钥来查询、修改、删除和插入数据,这样就要随时对数据库中数据进行加解密处理。因此,数据库密码系统宜采用公开密钥的加密方法。

3）多级密钥结构

数据库关系运算中参与运算的最小单位是字段，查询路径依次是库名、表名、记录名和字段名。因此，字段是最小的加密单位。也就是说当查得一个数据后，该数据所在得库名、表名、记录名、字段名都应是知道的。对应的库名、表名、记录名、字段名都应该具有自己的子密钥，这些子密钥组成了一个能够随时加/解密的公开密钥。

4）合理处理数据

这包括几方面的内容。首先要恰当地处理数据类型，否则DBMS将会因加密后的数据不符合定义的数据类型而拒绝加载，或因识别不了必须的部分数据无从完成对数据库文件的管理和使用；其次，需要处理数据的存储问题，实现数据库加密后，应基本上不增加空间开销。在目前条件下，数据库关系运算中的比较、匹配字段，如表间连接码、索引字段等数据不宜加密。据此，一般只能对数据库中数据进行部分加密。当然，从提高系统安全性出发还可选择联机全数据库加密或脱机全数据库加密等不同的加密方式。

5）不影响合法用户的操作

加密系统影响数据操作响应时间应尽量短。在现阶段，平均延迟时间不应超过0.1秒。此外，对数据库的合法用户来说，数据的录入、修改和检索操作应该是透明的，不需要考虑数据的加/解密问题。

3.1.5 数据仓库安全与数据库安全比较

数据仓库与数据库之间存在的差别使得数据仓库的安全性问题更加复杂，具体分析如下：

- 传统的操作型数据库的安全性是建立在应用程序层或者是数据库管理层

上的，是通过创建程序和事务来实现的。数据仓库中，在数据访问的范围内没有控制访问的应用程序基础，没有应用程序和系统管理层次上的访问控制[10]。

- 数据库的视图安全性机制不能满足数据仓库的需要[10]。数据库的安全性是建立在视图的定义和控制的基础上的。然而数据仓库环境和操作性环境不相同，用户的数量通常都是很大的，而且不规则的。难以在一个不知其用途的数据上面附加一个视图。此外视图还有一个很严重的局限性，就是当从磁盘装载数据时，所有视图的安全性问题就被旁路了。

- 数据库和数据仓库的数据模型不同。由于关系模型在操作型系统中是预先定义好的，而数据仓库系统中大多用的是非关系的多维模型，访问控制模型不是那么容易的匹配[16]。

- 数据仓库中受保护的对象不是表，而是维度、粒度层次等等。数据仓库中含有不同层次的数据，需要不同程度的安全性，越是高度概括性的数据需要的安全级别越高。

通过上面的分析可知:数据仓库在数据组织形式、使用目的上都与数据库有着很大的不同，同时数据对象比数据库中的表复杂的多，而且数据仓库的访问形式也多种多样，因此不能简单的把数据库中的安全技术拿到数据仓库上来用，数据仓库的安全性相对于数据库而言复杂的多。然而数据仓库中数据的增值又使得它对安全要求更加迫切，所有这些使得数据仓库的安全问题是一个值得研究的课题。

3.2 网络信息安全

随着现代通信技术的发展和迅速普及，特别是随着由通信与计算机相结合而诞生的计算机互联网络全面进入千家万户，使得信息共享应用日益广泛与深入。世界范围的信息革命激发了人类历史上最活跃的生产力，但同时也使得信息的安全问题日渐突出而且情况也越来越复杂。从大的方面来说，信息安全问题已威胁到国家的政治、经济、军事、文化、意识形态等领域。因此，“信息战”很早就被提出并将信息武器列为继原子武器、生物武器、化学武器之后的第四大武器。从小的方面来说，信息安全问题也是人们能否保护自己个人隐私的关键。

信息安全研究所涉及的领域相当广泛。从消息的层次来看，包括消息的完整性（即保证消息的来源、去向、内容真实无误）、保密性（即保证消息不会被非法泄漏扩散）、不可否认性（即保证消息的发送和接收者无法否认自己所做过的

操作行为）等。从网络层次来看，包括可靠性（即保证网络和信息系统随时可用，运行过程中不出现故障，若遇意外打击能够尽量减少损失并尽早恢复正常）、可控性（即保证营运者对网络和信息系统有足够的控制和管理能力）、互操作性（即保证协议和系统能够互相联接）、可计算性（即保证准确跟踪实体运行达到审计和识别的目的）等。从设备层次来看，包括质量保证、设备备份、物理安全等。从经营管理层次来看，包括人员可靠、规章制度完整等。如果再从行业层次来看，那所包含的内容就更无法穷尽了。比如：安全移动通信、安全数据通信、安全卫星通信、安全智能网、安全ISDN、安全计算机、安全网络、安全多媒体、安全HDTV、安全数据库、安全路由器、安全浏览器等等。由此可见，信息安全实际上是一门多学科的综合性研究课题，其边界几乎无法界定。下面将主要从技术角度重点介绍当今涉及通信安全和计算安全的几种常用而有效的信息安全方法。

3.2.1 互联网络的安全要求以及存在的安全威胁

总的来说，网络安全的五个基本的安全要求是:

1、机密性(Confidentiality):保证没有经过授权的用户，实体或进程无法窃取信息。在一个开放的网络环境里，维护信息机密是全面推广应用的重要保障。因此，要预防非法的信息存取和信息在传输过程中被非法窃取。

2、授权(Authorization):授权是确定允许用户做什么的过程。可将不同的特权给予不同类型的用户。

3、数据完整性(Data integrity):保证没有经过授权的用户不能改变或者删除信息，从而信息在传送的过程中不会被偶然或故意破坏，保持信息的完整、统一。因此，要预防对信息的随意生成、修改和删除，同时要防止数据传送过程中信息的丢失和重复。

4、原始性证明(Proof of Origin):对信息或数据的发送者的进行标识。保证信息被经过标识的发送者所传送，从而避免以前的数据包被重复发送。

5、防止抵赖(Nonrepudiation):保证信息的发送者不能抵赖或否认对信息的发送，当然信息发送前需要对发送者进行安全认证。要在信息的传输过程中为参与交易的个人、企业或国家提供可靠的标识。

最后的三个安全要求是彼此相关的，数据完整性与原始性证明的区别在于数据是完整的，并不能保证信息不被重复发送。换句话说，数据完整性不能防止反

复攻击。哈希散列算法如HMAC，认证时使用一个经过加密的密钥对于原始性证明来说是合适的，但并不适用于“防止抵赖”。

相应的，目前网络存在的安全威胁主要表现在以下几个方面：

1、非授权访问：没有预先经过同意，就使用网络或计算机资源被看作非授权访问。它主要有以下几种形式：假冒、身份攻击、非法用户进入网络系统进行违法操作、合法用户以未授权方式进行操作等。

2、信息遗漏丢失：指敏感数据在有意或无意中被泄漏出去或丢失。这种威胁主要来自窃听、搭线等信息探测攻击。

3、破坏数据完整性：以非法手段窃得对数据的使用权，删除、修改、插入或重发某些重要信息，以取得有益于攻击者的响应；恶意添加，修改数据，以干扰用户的正常使用。

4、拒绝服务攻击：是一种比较简单，但又日益流行的攻击和禁用企业信息资源的方法。在拒绝服务攻击中，作恶者发送大量的信息流量，使Web服务器、主机、路由器和其它网络设备负担过重。通过这种方式发送的信息流量非常之大，致使企业的用户、客户和合作伙伴都在好长一段时间内无法访问网络。

基于以上的安全威胁以及网络安全的迫切要求，仅仅依靠SSL的安全机制不能解决所有的问题。

3.2.2 网络安全体系

基于 Hurwitz Group[21]的五层网络安全体系，有的学者提出了六层网络安全体系[21]，即一套完整的网络安全解决方案需要从网络硬件设备的物理安全、网络传输的链路安全、网络级的安全、信息安全、应用安全和用户安全等 6 个方面综合考虑。

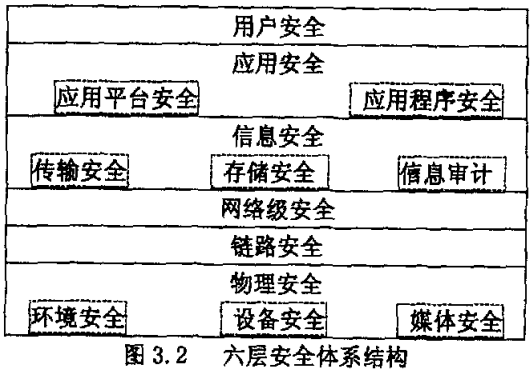


图 3.2　六层安全体系结构

物理安全，主要防止物理通路的损坏、物理通路的窃听、对物理通路的攻击（干扰）等。保证计算机信息系统各种设备的物理安全是整个计算机信息系统安全的前提，它通常包括环境安全（指系统所在环境的安全保护）、设备安全和媒体安全三个方面。抗干扰、防窃听将是物理层安全措施制定的重点。现在物理实体的安全管理现已有大量标准和规范，如计算机场地安全要求（GB9361-88）、计算机场地技术条件（GFB2887-88）等。

链路安全需要保证通过网络链路传送的数据不被窃听，主要针对公用信道的传输安全。在公共链路上采用一定的安全手段可以保证信息传输的安全，对抗通信链路上的窃听、窜改、重放、流量分析等攻击。在局域网内可以采用划分 VLAN（虚拟局域网）来对物理和逻辑网段进行有效的分割和隔离，消除不同安全级别逻辑网段间的窃听可能。如果是远程网，可以采用链路加密等手段。

网络级的安全需要从网络架构、网络访问控制、漏洞扫描、网络监控与入侵检测等多方面加以保证。首先要保证网络架构的正确，路由正确；采用防火墙、安全网关、VPN 等实施网络层的安全访问控制；此外可以采用漏洞扫描、网络监控与入侵检测系统等与防火墙结合使用，形成主动性的网络防护体系。

信息安全是一个重要的问题，它涉及信息传输安全、信息存储安全和信息审计等问题。保证信息传输安全需要保证信息的完整性、机密性、不可抵赖和可用性等；而对于信息存储安全，主要包括纯粹的数据信息和各种功能信息两大类，为确保这些数据的安全，可以采用数据备份和恢复、数据访问控制措施、数据机密性保护、数据完整性保护、防病毒、备份数据的安全保护等措施；此外，为防止与追查网上机密信息的泄漏行为，并防止不良信息的流入，可在网络系统与因特网的连接处，对进出网络的信息流实施内容审计。

应用层次的安全包括应用平台、应用程序的安全。应用平台的安全包括操作系统、数据库服务器、Web 服务器等系统平台的安全，由于应用平台的系统非常复杂，通常采用多种技术来增强应用平台的安全性。应用程序完成网络系统的最终目的一为用户服务，应用程序可以使用应用平台提供的安全服务来保证基本安全，如通信内容安全、通信双方的认证、审计等手段。

用户的安全性考虑的主要是用户的合法性，主要是用户的身份认证和访问控制。通常采用强有力的身份认证，确保密码难以被他人猜测到；并可以根据不同

的安全等级对用户进行分组管理，不同等级的用户只能访问与其等级相对应的系统资源和数据。

3.2.3 网络安全相关技术

1、身份认证

认证技术是信息安全理论与技术的一个重要方面。用户身份认证是系统防止非法用户侵入的第一道安全防线，它的目的是识别系统授权的合法用户，防止非法用户访问。单机状态下的用户登录计算机，一般有以下几种形式验证用户身份：

（1）用户所知道的东西，如口令、密码；

（2）用户所拥有的东西，如智能卡、身份证、护照、密钥盘；

（3）用户所具有的生物特征，如指纹、声音、视网膜扫描、DNA 等

网络环境下的身份认证较为复杂，主要是考虑到验证身份的双方一般都是通过网络而非直接交互，像根据指纹等手段就难以实现。同时大量的黑客随时随地都可能尝试向网络渗透，截获合法用户口令并冒名顶替以合法身份入网，所以目前一般采用高强度的密码认证协议技术来进行身份认证。主要有以下几种：

（1）一次性口令技术

（2）PPP（Point-to-point Protocol）中的认证协议

（3）远程拨入用户的鉴别服务RADIUS(Remote Authentication Dial-in User Service)

（4）Kerberos认证服务

（5）单点登陆系统(Single Sign On)

2、授权与访问控制

访问控制是通过某种途径显式地准许或限制用户访问能力或范围的一种方法。它是针对越权使用资源的防范措施，通过限制对关键资源的访问，防止非法用户的侵入或因为合法用户的不慎操作而造成的破坏，从而保证网络资源受控地、合法地使用。用户只能根据自己的权限大小来访问系统资源，不得越权访问。访问控制技术并不能取代身份认证，它是建立在身份认证的基础之上的，通俗地说，身份认证解决的是“你是谁，你是否真的是你所声称的身份”，而访问控制技术解决的是“你能做什么，你有什么样的权限”这个问题。

访问控制系统一般包括以下几个实体：

（1）主体（subject）:发出访问操作、存取要求的主动方，通常可以是用户或用户的某个进程等。

（2）客体（object）:被访问的对象，通常可以是被调用的程序、进程，要存取的数据、信息，要访问的文件、系统或各种网络设备、设施等资源。

（3）安全访问政策:一套规则，用以确定一个主题是否对客体拥有访问能力。

由此访问控制的目的可以阐述为：限制主体对访问客体的访问权限，从而使计算机系统在合法范围内使用；决定用户能做什么，也决定代表一定用户利益的程序能做什么。

常用的实现方法有访问控制矩阵、访问能力表、访问控制表和授权关系表等几种。在计算机系统中通常采用3种不同的访问控制策略：自主访问控制（DAC）、强化访问控制（MAC）和基于角色的访问控制（RBAC）。

自主访问控制DAC是目前计算机系统中实现最多的访问控制机制，它是在确认主体身份以及它们所属组的基础上对访问进行限定的一种方法，称其为自主型是因为在DAC系统中，一个拥有一定访问权限的主体可以直接或间接地将权限传给其他主体。其基本思想是：允许某个主体显式地指定其他主体对该主体所拥有的信息资源是否可以访问以及可执行的访问类型。

MAC主要用于多层次安全级别的军事应用当中，它预先将主体和客体分级，即定义用户的可信任级别及信息的敏感程度，然后根据主体和客体的级别标记来决定访问模式，用户的访问必须遵守安全政策划分的安全级别的设定以及有关访问权限的设定。当用户提出访问请求时，系统对主客体两者进行比较以确定访问是否合法。

RBAC根据用户在组织内所处的角色进行授权与访问控制。系统定义了各种角色，每种角色可以完成一定的职能，不同的用户根据其职能和责任被赋予相应的角色，一旦某个用户成为某角色的成员，则此用户可以完成该角色所具有的职能。

3、防火墙

防火墙是一种访问控制技术，在组织的网络和外界网络间设置障碍，阻止外界对内部资源的非法访问同时也防止内部对外部的不安全的访问。实现防火墙的主要技术有：数据包过滤、应用网关和代理服务。

(1)包过滤技术

包过滤技术，即在网络的适当位置对数据包实施有选择的通过。选择的依据是系统事先设定或随时定义的过滤逻辑(也称接入控制表)。通过检查数据流中的每个数据包，根据数据包的原地址、目的地址、端口号、TCP链路状态等要素或组合来限制数据包。只有满足过滤逻辑的数据包才被转发，否则被禁止。通过限制对特定端口的IP分组的禁止，可以防止黑客利用不安全的服务对内部网络进行攻击。

(2)应用网关技术

应用网关是建立在网络应用层上的协议过滤、转发技术，针对特别的网络应用协议指定数据过滤逻辑，并将数据包分析结果和采取措施进行登记和统计，形成审计报告。应用网关不像通用的机制允许不同类型的通信流通，而对每个应用采取专用的限制。该方法对每种应用均应该提供专门的用户程序和用户接口，工作量较大，不灵活而且效率较低，但比较安全。

(3)代理服务技术

代理服务是设置在Internet防火墙网关的专用应用级编码。包过滤和应用网关技术仅仅依据特定的逻辑检查。一旦特定的网络数据流满足逻辑，则防火墙内外的计算机系统建立直接联系，因而保留了防火墙外部的计算机系统直接了解内部网络结构和运行状态的可能。代理服务技术是针对该缺陷的挽救措施。防火墙内外计算机系统的“连接”由两个终止于代理服务的“连接”来实现，从而实现了外部计算机系统的隔离。代理服务的优点是，将被保护网络的内部结构屏蔽起来，同时实现较强的数据流监控、过滤、记录和报告功能。缺点是需要专门开发代理服务软件和相应的监控、过滤程序。

各种技术均有优缺点，现在的防火墙成熟产品多是将各种技术结合起来，如sunsoft公司的firewall-1实现了包过滤和应用网关技术。

4、信息加密

信息加密是保障信息安全的最基本、最核心的技术措施和理论基础。信息加密也是现代密码学的主要组成部分。信息加密过程由形形色色的加密算法来具体实施，它以很小的代价提供很大的安全保护。在多数情况下，信息加密是保证信息机密性的唯一方法。据不完全统计，到目前为止，已经公开发表的各种加密算法多达数百种。如果按照收发双方密钥是否相同来分类，可以将这些加密算法分

为常规密码算法和公钥密码算法。

在常规密码中，收信方和发信方使用相同的密钥，即加密密钥和解密密钥是相同或等价的。比较著名的常规密码算法有：美国的DES及其各种变形，比如Triple DES、GDES、NesDES和DES的前身Lucifer;欧洲的IDEA;Skipjack、RC4、RC5以及以代换密码和转轮密码为代表的古典密码等。在众多的常规密码中影响最大的是DES密码。

DES由IBM公司研制，并于1977年被美国国家标准局确定为联邦信息标准中的一项。ISO也已将DES定为数据加密标准。DES是世界上最早被公认的实用密码算法标准，目前它已经受住了长达20年之久的实践考验。DES采用56比特长度的密钥将64比特长的数据加密成等长的密文。在DES的加密过程中，先对64比特长的明文块进行初始置换，然后将其分割成左右各32比特长的子块，经过16次迭代，进行循环移位与变换，最后再进行逆变换得到64比特长的密文。DES的解密过程与加密过程很相似，只需将密钥的使用顺序进行颠倒。DES算法采用了散布、混乱等基本技巧，构成其算法的基本单元是简单的置换、代替和模2加法。DES的整个算法结构都是公开的，其安全性由密钥保证。DES的加密速度很快，可用硬件芯片实现，适合于大量数据加密。

在公钥密码中，收信方和发信方使用的密码互不相同，而且几乎不可能由加密密钥推导出解密密钥。比较著名的公钥密码算法有：RSA、背包密码、McEliec密码、Diffle Hellman、Rabin、Ong Fiat Shamir、零知识证明的算法、EllipticCurve、ElGa mal算法等等。最有影响的公钥加密算法是RSA，它能够抵抗到目前为止已知的所有密码攻击。RSA的优点是不需要密钥分配，但缺点是速度慢。

当然在实际应用中人们通常是将常规密码和公钥密码结合在一起使用，比如：利用DES或者IDEA来加密信息，而采用RSA来传递会话密钥。如果按照每次加密所处理的比特数来分类，可以将加密算法分为序列密码和分组密码。前者每次只加密一个比特，而后者则先将信息序列分组，每次处理一个组。

5、信息确认

信息确认技术通过严格限定信息的共享范围来达到防止信息被非法伪造、窜改和假冒。一个安全的信息确认方案应该能使：①合法的接收者能够验证他收到的

消息是否真实；②发信者无法抵赖自己发出的消息；③除合法发信者外，别人无法伪造消息；④发生争执时可由第三人仲裁。按照其具体目的，信息确认系统可分为消息确认、身份确认和数字签名。消息确认使约定的接受者能够验证消息是否是约定发信者送出的且在通信过程中未被篡改过的消息。身份确认使得用户的身份能够被正确判定。最简单但却最常用的身份确认方法由：个人识别号、口令、个人特征（如指纹）等。数字签名与日常生活中的手写签名效果一样。它不但能使消息接受者确认消息是否来自合法方，而且可以为仲裁者提供发信者对消息签名的证据。

用于消息确认中的常用算法有：ElGamal签名、数字签名标准（DSS）、One time签名、Undeniable签名、Fail stop签名、Schnorr确认方案、Okamoto确认方案、Guillou Quisquater确认方案、Snefru、N hash、MD4、MD5等等。其中最著名的算法也许是数字签名标准（DSS）算法。

6、审计技术

它使信息系统自动记录下网络机器的使用时间、敏感操作和违纪操作等。审计类似于飞机上的“黑匣子”，它为系统进行事故原因查询、定位、事故发生前的预测、报警以及为事故发生后的实时处理提供详细可靠的依据或支持。审计对用户的正常操作也有记载，因为往往有些“正常”操作（如修改数据等）恰恰是攻击系统的非法操作。

7、安全协议

整个网络系统的安全强度实际上取决于所使用的安全协议的安全性。安全协议的设计和改进有两种方式：

①对现有网络协议（如 TCP/IP）进行修改和补充

②在网络应用层和传输层之间增加安全子层，如安全协议套接字层（SSL），安全超文本传输协议（SHTTP）和专用通信协议（PCP）。安全协议实现身份鉴别、密钥分配、数据加密、防止信息重传和不可否认等安全机制。

第四章 XML及相关安全技术

XML(eXtensible Markup Language)是由世界万维网联盟W3C制定的一种数据标准，它和HTML都是由SGML衍生的。但是XML有其自身的优点，它比HTML的功能强大，比SGML简单，XML以其结构化、标签化、易于交换和灵活性在很多行业得以广泛的应用。XML已逐步成为Internet交换数据的一种机制，因此XML在存储和交换时的安全变的非常重要。目前，与XML相关的安全性规范主要有XML加密、XML签名、XML密钥管理（XKMS）、XML访问控制语言（XACL）等。

4.1 XML加密

XML加密是一个对明文加密产生密文以及对密文解密恢复明文数据的过程。它可以用来保证数据的机密性。XML加密(XML Encryption)为需要安全地交换结构化数据的应用程序提供了端到端的安全服务。XML本身就是结构化数据最流行的技术，因此基于XML的加密技术自然成为满足数据交换应用程序的复杂的安全性能要求的技术。

XML文档可以和任何其他文档一样作为一个整体进行加密，然后安全地发送给一个或多个接收者。这是SSL或TLS的一个普通的功能。但是，更为有趣的是，如何来解决这种情形:加密XML文档的某些部分，而让那些不包含敏感信息的部分以明文的形式存在。XML Encryption的一个重要的特点就是它支持对XML文档的一些特定部分进行加密。对于同一个文档中的不同部分用不同的密钥进行加密，就可以把同一个XML文件发给不同的接收者，而接收者只能看见和他相关的部分。

加密的数据可以是任意的数据(包括XML文档)， XML元素，或者XML元素内容。加密数据的结果是一个XML加密元素<EncryptedData>，它包含或引用密文数据。这个标准使XML数据提供者可以根据用户的不同对内容进行颗粒化的控制。而且，由于数据本身而不是整个文件是加密的，整个文件还是可以被XML处理器识别和处理。

1、加密的粒度

直接对文档加密遇到的问题就是加密的粒度太大，使得有的要求无法满足，而且影响效率。W3C对加密的粒度进行了分类，包括：

- 单个XML元素的加密；
- 多个XML元素的加密；
- XML元素内容的加密；
- XML文档的加密
- 对加密信息的加密

2、加密的语法

W3C标准定义了很多加密后产生的元素格式，包括：

- EncryptedType，由EncryptedType抽象类型派生出EncryptedData和EncryptedKey；
- EncryptionMethod元素，它是用于描述密文数据的加密算法的；
- CipherData元素，它是用于表示加密后的数据的；如果加密的数据不是直接表示，可以使用CipherReference元素，它可以指向一个数据源，该数据源产生加密过的数据；
- EncryptedData元素，它是核心元素。不仅其子元素CipherData包含加密过的数据，而且它代替了被加密的数据，或者是加密后document的根元素；
- Ds:KeyInfo元素，它是用来描述密钥获取方式的，一般有三种方式：

  （1） EncryptedData或EncryptedKey中的子元素ds:KeyInfo获得密钥。不建议使用ds:KeyValue明文表示密钥；建议使用ds:KeyName指向EncryptedKey CarriedKeyName；建议使用ds:RetrievalMethod描述密钥获取方法；EncryptedKey元素，它是用来从发送方传递密钥给接受方的；ds:RetrievalMethod用来提供EncryptedKey元素链接；ReferenceList是包含从密钥指向用该密钥加密数据的链接集合；

  （2） 使用DataReference或者KeyReference指定密钥；

  （3） 通过接受方的应用程序获得密钥

- EncryptedProperties元素，关于产生EncryptedData或者EncryptedKey的额外相关信息

3、加密步骤

W3C描述了使用该标准的加密步骤：

（1） 选择加密的算法；

（2） 选择加密的密钥，如果密钥是通过链接指定的，构造ds:KeyInfo；如果密钥本身也是被加密的，通过递归的调用本步骤构造EncryptedKey；

（3） 加密数据。如果数据是XML的Element或者Element Content，数据应该序列化成UTF-8字节流；如果数据不是字节流，应用程序必须把数据先序列成字节流；如果数据是字节流，使用上述步骤1，2来加密；

（4） 构造EncryptedType，EncryptedType包括了加密算法，参数，密钥和加密数据的类型，如果加密的字节流放入EncryptedData元素中的CipherData中，那么加密的字节流是base64编码的；如果加密的字节流没有放入EncryptedData元素，在CipherReference中放入URI或者其他可以得到加密数据的链接；

（5） EncryptedData的处理，加密程序必须返回EncryptedData元素给应用程序，应用程序可以使用该EncryptedData作为根元素创建新的XML文档，或者嵌入其他的XML文档中。

4、W3C相应的解密步骤：

（1） 处理element得到算法，参数和ds:KeyInfo元素，如果有些数据被忽略，应用程序必须提供；

（2） 根据ds:KeyInfo元素，得到加密密钥；

（3） 解密CipherData元素中的数据，如果提供一个CipherValue子元素，对其内容base64解码，得到加密后的字节流；如果提供CipherReference子元素，通过Reference得到加密后的字节流；使用步骤1，2获得的算法，参数和密钥解密加密过的字节流；如果密钥值是明文，密钥被保存并用来解密EncryptedData元素，解密方必须把密钥值传递给应用程序保存；

（4） 处理Element或者Element content的加密数据类型：把第三步得到的明文字节流转化成UTF-8编码的字符数据；解密者必须能够返回类型和UTF-8编码的XML字符数据；解密者应该提供解密结果替换解密前的元素的功能；

（5） 如果不清楚解密后的数据类型，或者解密后的类型不是Element或者Element Content，把经第3步解密的输出返回给应用程序作进一步处理。

5、支持的算法：

W3C的XML Encryption讨论了如下的加密算法：

- 块加密。它是针对加解密固定长度，多块字节流数据的算法，有3DES，AES128，AES256，AES-192（可选的）算法；
- 流加密，最简单的流加密就是使用产生的字节流与明文异或操作，生成密文，支持用户指定的算法；
- Key Transport:是用于密钥传递的算法，有RSA－v1.5，RSA-OAEP算法；
- Key Agreement:是一种根据发送方和接受方公钥进行计算，得到共享的密钥，有Diffie-Hellman（可选的）算法；
- Symmetric Key Wrap:是针对对称key加解密的算法，有3DES KeyWrap，AES128，AES256，AES-192（可选的）算法；
- Message Digest:作为导出key的一部分用在AgreementMethod中，有SHA1,SHA256(建议)，SHA512(可选的)，RIPEMD-160(可选的)算法；
- Message Authentication:提供key的基本鉴定，使用XML Digital Signature(建议)；
- Canonicalization:是一种在加密之前把XML转化为字节流的方法，包含Canonical XML(可选的)，Canonical XML with comments(可选的)，Exclusive XML Canonicalization(可选的)，Exclusive XML Canonicalization with Comments几种情况；
- Encoding:使用base64进行编码。

6、加密示例：

XML Encryption规范给出了根据不同的需求对不同的对象加密的例子。下面就是一些XML加密的简单的例子。

清单 1. 显示 John Smith 的银行帐户、5000 美元限额、卡号和有效期的信息

```
<?xml version='1.0'?>
<PaymentInfo xmlns='http://example.org/paymentv2'>
    <Name>John Smith</Name>
     <CreditCard Limit='5,000' Currency='USD'>
      <Number>4019 2445 0277 5567</Number>
      <Issuer>Example Bank</Issuer>
     <Expiration>04/02</Expiration>
```

```
        </CreditCard>
    </PaymentInfo>
```

清单 2. 除名称之外全部被加密的加密文档

```
    <?xml version='1.0'?>
      <PaymentInfo xmlns='http://example.org/paymentv2'>
        <Name>John Smith<Name/>
        <EncryptedData Type='http://www.w3.org/2001/04/xmlenc#Element'
         xmlns='http://www.w3.org/2001/04/xmlenc#'>
      <CipherData><CipherValue>A23B45C56</CipherValue></CipherData>
        </EncryptedData>
      </PaymentInfo>
```

清单 3. 只隐藏了信用卡号的加密文档

```
      <?xml version='1.0'?>
       <PaymentInfo xmlns='http://example.org/paymentv2'>
         <Name>John Smith<Name/>
         <CreditCard Limit='5,000' Currency='USD'>
           <Number>
             <EncryptedData xmlns='http://www.w3.org/2001/04/xmlenc#'
              Type='http://www.w3.org/2001/04/xmlenc#Content'>
                 <CipherData><CipherValue>A23B45C56</CipherValue>
                 </CipherData>
             </EncryptedData>
           </Number>
           <Issuer>Bank of the Internet</Issuer>
           <Expiration>04/02</Expiration>
         </CreditCard>
      </PaymentInfo>
```

清单 4. 隐藏了全部内容的加密文档

```
<?xml version='1.0'?>
```

```
<EncryptedData xmlns='http://www.w3.org/2001/04/xmlenc#'
Type='http://www.isi.edu/in-notes/iana/assignments/media-types/text/xml'>
<CipherData><CipherValue>A23B45C56</CipherValue></CipherData>
</EncryptedData>
```

加密后的XML文档一定不要包含对DTD的引用，否则会使攻击者获知部分明文，对明文、密文进行分析，可能会破译密文，获得秘钥。

7、实现方案

作为XML加密的用户，可能更多的使用XML加密的某个实现工具，而不是直接使用规范的语法。

XML加密工具包：该组实现工具对加密和解密XML以及任意内容提供了一个程序化的API和算法上的支持，以下列举了其中一些加密工具包：

- IBM XML Securiy Suite
- XML Security Library
- Phaos XML toolkit
- Baltimore

4.2 XML签名

XML签名使得Web服务成为现实，它是Web服务的基础。XML签名是另一个W3C的推荐标准。和安全认证签名相似，XML签名也是用于确保XML文件内容没有被篡改的。为了适应各种文件系统和处理器在版式上的不同，XML签名采用了“规范化(canonicalization)”。这就使得XML签名可以适应XML文件可能遇到的各种环境。当对内容进行签名时，规范化使文件里的数据和标识产生一个独一无二的签名，忽略了一些诸如段落结束或者制表符之类的次要信息。

XML签名和XML加密紧密相关。XML签名和XML加密结合在一起，可以确保数据发送和接收的一致性。

XML签名的安全模型是公钥签名，先进行hash计算，再对摘要值进行签名。对证书和秘钥的验证(Validity)由XKMS实现。

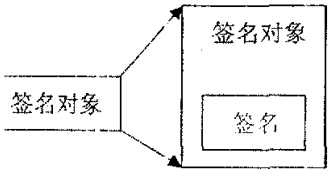
现在已有开发工具包有：

- IBM alphaWorks XML Security Suite

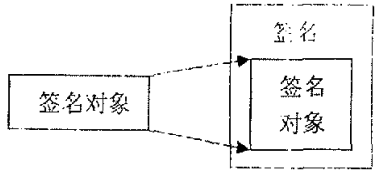
- IAIK IXSIL
- VeriSign Trust Services Integration Kit

1、签名的种类

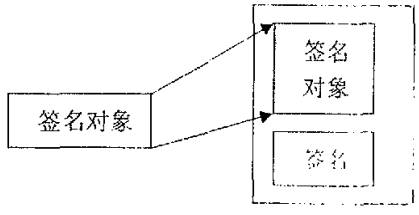
根据XML文档正文与签名数据之间的关系，可以对签名进行分类：



(a)Enveloped Signature



(b)Enveloping Signature



(c)Detached Signature

图4.1 签名的种类

- Enveloped Signature

签名对象中包含签名，如图4.1（a）所示；

- Enveloping Signature

签名中包含签名对象，如图4.1（b）所示；

- Detached Signature

生成的签名和签名对象不存在包含关系，使用起来很方便，如图4.1（c）所示。

2、签名的产生

根据W3C的XML签名规范，XML签名产生的步骤如下：

（1）生成签名对象的信息：

- 转换签名对象；
- 生成签名对象的Digest值；
- 生成签名对象的信息；

（2）生成签名信息：

- 使用正规化（Canonicalize）方法和签名值生成算法（如DSS）做成签名信息；
- 根据指定的算法生成签名值；
- 把上述使用的签名信息，签名对象，密钥信息，签名值构造成最终的XML签名。

3、签名的验证

根据W3C的XML签名规范，XML签名验证的步骤如下：

（1）签名对象信息的验证：

- 签名信息的正规化；
- 生成签名对象的Digest值；
- 生成的Digest值与记述的Digest值进行比较；

（2）签名信息的验证：

- 从提供的信息中获取密钥；
- 对签名值做成算法进行正规化；
- 根据签名信息验证签名值。

4、签名的语法

W3C标准定义了很多签名后产生的元素格式，包括：

- Signature元素，是XML签名的根元素；
- SignatureValue元素，该元素包括签名的实际值，一般情况下是经过base64编码的；
- SignedInfo元素，该结构中包含了正规化算法，签名算法，一个或多个Reference。其中CanonicalizationMethod元素，用来指定在应用签名算法之前，对SignedInfo元素使用的正规化方法；SignatureMethod元素，是用来指定签名产生和验证的算法；Reference元素，是可能出现一次或多次的元素，它指定了Digest算法和Digest值，也可以是一个指向签名对象的标识符和该对象的类型，可以包括在Digest之前转换的列表；
- KeyInfo元素，是一个可选的元素，接受者可以根据它获取用来验证签名的密钥；其中包括KeyName元素，KeyValue元素，RetrievalMethod元素，X509Data元素，PGPData元素，SPKIData元素，MgmtData元素；

- Object元素，是一个可选的元素，它可以包含任何的数据；
- Manifest元素，它提供一组Reference；
- SignatureProperties元素，关于签名产生的额外信息可以记录在该元素中；

5、支持的算法

W3C的XML签名规范中讨论了如下的签名算法：

- 消息散列：SHA-1算法
- 消息验证码：HMAC算法
- 签名算法：DSA算法，PKCS1(RSA-SHA1)算法；
- 正规化（Canonicalization）算法：忽略comments的正规化XML算法，包含comments的正规化XML算法；
- 转换算法：正规化（Canonicalization）算法，Base64编码，Xpath处理算法，Enveloped Signature转化算法，XSLT转化算法。

6、XML签名举例

XML签名支持检测数据的完整性，消息认证和签名者身份认证，这些服务适用于任何数据类型，不管是位于包含签名的XML文档中还是其他地方。XML签名适用于任何数字内容(或称为数据对象)，包括XML。一个XML签名可应用于一个或多个资源的内容。下面举一个XML签名的例子：

XML文档中的PaymentInfo元素：

```
<?xml version=' 1.0'?>
<!DOCTYPE PaymentInfo SYSTEM "PaymentInfo.dtd">
    <PaymentInfo Id='data'
       xmlns='htp://example.orgipaymentv2'>
         <Name>John Smith<、Name>
         <CreditCard Limit='5,000' Curency='USD'>
         <Number>4019 2445 0277 5567</Number>
         <Issuer>Example Bank</Issuer>
         <Expiration>04/02</Expiration>
    </CreditCard>
```

```
</PaymentInfo>
```

若对以上的PaymentInfo元素进行签名，得到的封装式签名如下所示：

```
[01] <Signature xmlns='http://www.w3.org/2000/09/xmidsig#'>
[02] <SignedInfo>
[03]<CanonicalizationMethod
Algorithm='http://www.w3.org/TR/2001/REC-xml-c14n-20010315'/>
[04] <SignatureMethodAlgorithm='http://www.w3.org/2000/09/xmidsig#dsa-sha1'/>
[05] <Reference URI='#data'>
[06]      <Transforms>
[07] <TransformAlgorithm='http://www.w3.org/TR/2001/REC-xml-c14n-20010315'/>
[08]     </Transforms>
[09] <DigestMethodAlgorithm='http://www.w3.org/2000/09/xmldsig#sha1' />
[10]         <DigestValue>j61wx3rvEP...</DigestValue>
[11]   </Reference>
[12] </SignedInfo>
[13] <SignatureValue>...</SignatureValue>
[14] <KeyInfo>
[15]    <KeyValue>
[16]   <DSAKeyValue><P>…</P><Q>…</Q><G>…</G><Y>…</Y></DSAKeyValue>
[17] </KeyValue>
[18] </Keyinfo>
[19] <Object>
[20] <PaymentInfo Id='data' xmlns='http://example.org/paymentv2'>
[21] <Name>John Smith</Name>
[22] <CreditCard Limit='5,000' Curency='USD'>
[23] <Number>4019 2445 0277 5567</Number>
[24] <Issuer>Example Bank</issuer>
[25] <Expiration>04/02</Expiration>
[26] </CreditCard>
```

[27] </PaymentInfo>

[28] </Object>

[29] </Signamre>

[01]一[29）行的Signature元素包含了XML文档的 PaymentInfo元素（在[20]-[27]行）。实际上签名的信息在[02]一[12]行之间，即Signedlnfo元素。在[04]行的SignatureMethod元素引用的是将规范化(canonicalized)的SignedInfo转化成SignatureValue的签名算法。在签名部分中(SignedInfo元素中)包含用于计算SignatureValue元素的算法的引用，而SignatureValue元素本身却在SignedInfo元素外，在[13]行。[05]行的Reference元素的这个可选URI属性标识要签名的源数据对象，即[20]一[27]行的Paymentlnfo元素。[14]一[18]行的KeyInfo元素(该元素可选)指定用来验证签名的密钥。对KeyInfo元素的处理通常XKMS。SignedInfo的核心验证由两个必要过程组成:对Signedlnfo的签名验证和SignedInfo内部每个Reference摘要的验证。

由上面的例子我们来看XML签名和验证的过程，它与普通的签名类似。

签名的过程:

1、生成references

- 对源数据按照<Transforms>进行转换
- 计算摘要值放入<DigestValue>
- 生成<Reference>元素

2、生成Signature

- 将<Reference>元素置入<SignedInfo>元素之内
- 对<SignedInfo>元素计算<SignatureValue>值
- 构建包含<SignedInfo>的<Signature>元素

验证的过程:

1、验证references

- 对源数据按照<Transforms>进行转换
- 计算摘要值，与<DigestValue>进行比较

2、验证签名

- 从<KeyInfo>或其他资源获取秘钥

- 验证<SignatureValue>

可以看出签名的过程和验证签名的过程中生成和验证references是基本相同的。

4.3 XML密钥管理规范（XKMS）

为了简化PKI和数字证书与XML应用程序的集成，W3C制定了一项新标准XKMS。它定义了分发和注册XML签名所使用的公钥的方法，有待于同XML加密联合使用。XKMS包含两部分：XML密钥信息服务规范X-KISS和XML密钥注册服务规范X-KRSS。X-KRSS是用于注册公共密钥的，而X-KISS是用在为XML签名提供密钥信息服务方面。

- XML密钥信息服务规范（X-KISS）：X-KISS定义了信任服务的协议，它支持应用程序把对于XML Signature, XML Encryption或其他公钥关联的密钥信息的处理任务委托给一个信任服务。它的功能包括定位请求的公钥和绑定该密钥和标识符信息。它解决了在XML Signature中<KeyInfo>元素包含的公钥信息的问题。
- XML密钥注册服务规范（X-KRSS）：该协议支持密钥对持有者向信任服务系统注册密钥对，也就是公钥信息，从而该密钥对随后可以与XML Key Information Service Specification或更高层的信任断言服务，例如XTASS联合使用。

4.4 XML访问控制语言（XACL）

对于保存在本地或服务器端的XML文档，一种有效的安全方法就是访问控制。目前较为成熟的访问控制语言即XACL，它为XML提供了一种成熟的访问控制机制。与其它策略语言相似，XACL也是面向“对象一主体一操作一条件”的语言。主体可以是一个三元组(用户ID、角色和团体)，角色和团体都采用层次结构。对象支持从整个XML文档中到其中的单个元素。基本操作有读、写、创建和删除。

XACL利用XML文档及主体的层次结构特点，定义了授权传递策略和冲突解决策略。即对某一对象定义的授权可以沿主体、客体的层次结构向上或向下传递，并使用了传递选项来控制授权传递的深度。由于授权传递必须支持特例，可对某对象直接定义肯定或否定授权，因此在实现时可能会出现同一主体对同一对象具有相矛盾的授权，它可能是直接的，也可能是由于传递导致的，此即授权冲突。XACL

也定义了相应的冲突解决策略。

XACL采用临时授权模型，如图4.2所示，它主要由访问评估模块和请求执行模块组成。该模型定义了临时操作，即系统执行某基本操作时，可能要求执行相应的临时操作，如审计、日志、数字签名、认证等。相关策略的实现可分四个步骤：(1)提交请求：包括主体、对象和操作。(2)访问评估：根据目标XML文档的相关策略和相关状态来评价访问请求。访问决定要表明允许或拒绝，还要指定附加的临时操作。(3)在请求执行模块中处理请求：执行访问决定中指定的基本操作和附加的临时操作。对非“读”操作，目标XML文档及相关的状态文件将被更新。(4)返回视图：对“读”操作，创建并返回对象视图，其中只包含允许请求者访问的节点。

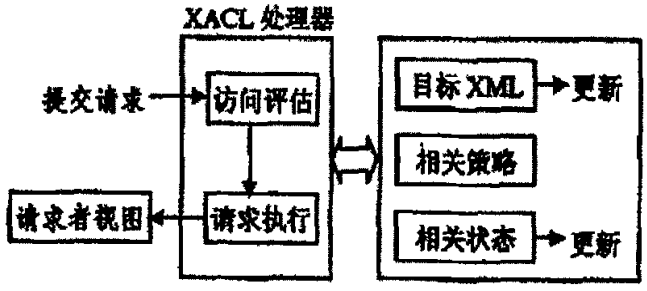


图4.2 临时授权模型

4.5 简单对象访问协议（SOAP）

简单对象访问协议SOAP (Simple Object Access Protocol)是在分散或分布式的环境中，通过XML交换信息的一种简单协议。

当前有多种创建应用程序的平台。但每种平台都习惯于使用自身的协议(本质上通常是二进制代码)来实现机器间的集成。因此，跨平台的应用程序在数据共享方面的能力相当有限。认识到这些限制后，人们一直在致力于建立有关数据格式和数据交换方面的标准，藉此实现“不论服务采用何种软件，使用何种硬件，都能够跨越这一传统的界限，以Web的形式无缝的将它们集成在一起”这一远景目标。目前，这一目标已迅速发展成为一种新的计算范例。该目标的核心是互操作性概念，即不同系统能够无缝的进行通信和共享数据。这也是Web服务追求的目标。Web服务是一种可以用标准Internet协议来访问的可编程应用逻辑;从另一个角度来

说，Web服务是有关机器间和应用程序间透明通信的、借助于Web的标准的具体实现。而SOAP，正是实现Web服务的核心协议之一。

SOAP为在一个松散的、分布的环境中使用XML对等的交换结构化的和类型化的信息提供了一个简单的轻量级机制。SOAP本身并不定义任何应用语义，如编程模型或特定语义实现，它只是定义了一种简单的机制，通过一个模块化的包装模型和对模块中特定格式编码的数据重编码机制来表示应用语义。SOAP的这项能力使得它可被很多类型的系统用于从消息系统到RPC (Remote Procedure Call)的延伸。SOAP也没有定义任何底层的传输协议，尽管在大部分情况下，SOAP被“默认”绑定到HTTP协议;但也可以用HTTP, FTP, SMTP甚至是JMS，或是您自己定义的某种P2P协议，来传输SOAP消息。

从某种意义上说，SOAP可以简单理解为:HTTP+XML+RPC:采用HTTP作为底层通信协议，RPC作为一致性的调用途径，XML作为数据传送的格式，允许服务提供者和服务客户经过防火墙在internet进行通信交互。SOAP被设计的一个主要目的，就是为了实现基于XML的RPC (Remote Procedure Call，远程过程调用)。尽管HTTP不是有效率的通信协议，而且XML还需要额外的文件解析，两者使得交易的速度可能大大低于其他方案;但是XML是一个独立于编程语言的、开放、健全、有语义的信息机制，而HTTP是一个广泛使用的协议，而且能避免许多关于防火墙的问题，从而使SOAP得到了广泛的应用。SOAP的两个主要设计目标是简单性和可扩展性。这就意味着有一些传统消息系统或分布式对象系统中的某些性质将不是SOAP规范的一部分.比如:分布式垃圾收集(Distributed garbage collection)、成批传送消息(Boxcarring or batching ofmessages)、对象引用[Objects-by-reference (which requires distributed garbagcollection)]、对象激活[Activation (which requires objects-by-reference)].

SOAP消息由一个强制的SOAP Envelope、一个可选的SOAP Header和一个强制的SOAP Body组成的XML文档。SOAP消息应当包含如下部分:

- 一个SOAP Envelope. Envelope是表示该消息的XML文档的顶级元素。
- 一个SOAP Header. Header是为了支持在松散环境下在通信方(可能是SOAP发送者、SOAP接收者或是一个或多个SOAP的传输中介)之间尚未预先达成一致的情况下为SOAP消息增加特性的通用机制，这通常应用在认证、事务管理以及支

付等应用中。SOAP定义了很少的一些属性用来指明谁可以处理该特性以及它是可选的还是强制的。

- 一个SOAP Body. Body为该消息的最终接收者所想要得到的那些强制信息提供了一个容器，即它包含了SOAP消息的实际内容。此外，SOAP定义了Body的一个子元素Fault用于报告错误。

通常，刚接触SOAP的用户提出的第一个问题就是SOAP如何解决安全性问题。在其早期开发阶段，SOAP被看作是基于HTTP的协议，所以认为HTTP的安全性对于SOAP已经足够了。毕竟目前有数以千计的Web应用程序都在使用HTTP 安全性，所以这对于SOAP确实已经足够。因此，当前的SOAP标准假定安全性属于传输问题，而并不作为安全性问题处理。当SOAP扩展至更为通用的协议，并运行于众多传输之上时，安全性问题就变得突出了。例如，HTTP提供若干种方法对进行SOAP调用的用户进行身份验证，但是当消息从HTTP路由到SMTP传输时，怎样传播该身份标识呢?SOAP是作为构造块协议进行设计的，所以幸运的是，已经有了相应的规范以基于SOAP为Web服务提供额外的安全保护功能。WS-Security规范定义了一套完整的加密系统，而WS-License规范定义了相应的技术，以保证调用者的身份标识，并确保只有授权用户才可以使用Web服务。

4.6 XML防火墙

1、使用XML防火墙的必要性

Web Services架构的特点以及新的业务需求引起一些新的安全问题，传统的网络防火墙(network firewall)难以解决。这里所说的网络防火墙是指工作在网络层的传统的防火墙。它通过IP地址、端口以及相应的安全策略来过滤数据包，将非法的和不安全的访问拦截在内网之外，但是它面临以下问题：

① 大部分的攻击、破坏和安全隐患来自防火墙所保护的网络内部，而不是外部。网络防火墙无法阻止网络内部对Web Services服务的非法访问和攻击。

② Web Services使用SOAP(Simple Object Access Protocol)协议进行通信。SOAP实质是一种使用XML传送结构化和类型化数据的小型协议。SOAP消息的设计很容易被利用穿透现有防火墙。

③ 在某些B2B业务中，需要进行跨防火墙的整合应用。

④ 同一个Web Services服务器上部署了很多不同安全等级的Web service，

同一个Web Services的不同方法也具有不同的安全等级，同一个Web Services的同一方法对不同的访问者具有不同的机密等级。在这种情况下，传统的基于IP地址和端口过滤数据包的网络防火墙是无法满足这种需求的，而XML防火墙较好地解决上述问题。XML防火墙的出现打破了应用层安全这个瓶颈。

2、XML防火墙的工作原理和作用

XML防火墙是一种应用防火墙，工作在应用层(applicaion-level)[35]。它对请求Web服务的SOAP消息进行深度解析，分析其中的安全信息，包括判断消息的来源，请求者的身份和权限，SOAP消息的机密性和完整性，欲访问的web service及其方法的安全等级，运用设置的安全策略、访问列表和知识库对之进行验证和检测，拒绝非法的、不安全的访问，为WebServices应用提供灵活的、丰富的安全解决方案。XML 防火墙运用了WS-Security规范 、XML签名、XML加密以及SAML(Security As—sertion Markup Language，安全声明标记语言) 等XML安全规范。首先，XML防火墙的一个基本功能就是认证和访问控制。包括：确认请求Web Services的实体(人、程序或者机构)的身份、识别请求者拥有的访问权限等级。其次，XML防火墙能够确保消息请求和消息回应的机密性、完整性和不可抵赖性。使用传输层加密SSL、XML签名和XML加密技术实现这个功能。此外，XML防火墙可以提高SOAP的服务质量(QOS)；可以抵御密码字典攻击、拒绝访问式攻击(DoS)、XML Response攻击等等。XML防火墙和传统的网络防火墙可以整合起来协同工作，在网络防火墙所保护的网络区域内部，再使用XML防火墙保护Web Services服务。

3、XML防火墙的特点

XML防火墙具有灵活性强、易于扩展等特点，用户可以根据自己的需要构建灵活的安全解决方案。下一代的Web Service具有链式结构和事务性特征，构建传统的防火墙代价太大，而使用XML防火墙能够高效的、更好地保护Web Services的安全。

4、XML防火墙与传统网络防火墙的比较

XML防火墙与传统的网络防火墙在工作原理，保护的对象，控制访问的策略等方面存在着很大的差异。从以下几个方面加以分析：从功能原型来看，网络防火墙决定是否让一个数据包路由到特定IP地址的特定端口．而XML防火墙决定是否让一个SOAP消息访问一个特定的Web Services方法。从控制的对象来看，网络防火

墙监控和分析IP地址、端口、协议、数据包等对象，而XML防火墙监控和分析的对象是Application(Web Services)、API(Operations)、请求服务的用户、XML / SOAP消息、URLs。从授权访问的依据来看，网络防火墙把IP地址、端口，以及访问流是否来自安全的网络区域等因素作为授权访问或者拒绝访问的依据。XML防火墙则是根据Web服务请求者的身份、角色，Web Services方法的类别以及基于角色的访问控制策略等因素，决定是授权访问还是拒绝访问。从信息传输安全来看，网络防火墙加密整个数据，而XML防火墙加密信息的粒度可以选择，可以加密整个SOAP消息，也可以加密SOAP消息的部分元素或者元素的内容。

第五章 WEB环境下的数据仓库的安全研究

5.1 WEB环境下数据仓库安全的主要内容

从WEB环境下的数据仓库的体系结构和资源组成来分析，数据仓库的安全应包括以下5个方面的内容：

①实体安全。系统设备及相关设施运行正常，服务适时。其具体包括环境、设备、机房、电磁辐射和数据介质等安全。

②数据安全。指系统拥有的和产生的数据或信息，如底层信息源中的数据、程序运行时得到的数据等完整、有效、使用合法和不被破坏或泄露。具体包括数据的输入、输出、存取控制、加密、备份与恢复等。

③软件安全。软件是数据仓库系统工作的主要平台，它的安全是数据仓库安全的重要内容，是研究的重点。包括操作系统、数据库管理系统、网络软件、应用软件及相关资料的完整和安全。

④运行安全。系统资源使用合法，包括电源、数据与介质管理、机房管理、运行管理和维护。

⑤网络安全。网络安全是指为保证网络及其节点安全而采用的技术和方法。它主要包括报文鉴别技术；数字签名技术；访问控制技术；数据加密技术；密钥管理技术；保证线路安全、传输安全而采用的安全传输介质；网络监测、跟踪及隔离技术；路由控制和流量分析控制技术等。

5.2 WEB环境下的数据仓库存在的安全隐患

数据仓库中存储着丰富的数据。另外，数据仓库技术使得数据仓库中的数据的价值得到了提升。数据从数据源中经过抽取、清洗、重新组织等过程才进入到数据仓库系统中，这些经过整理的数据在数据仓库中已经增值。存储在数据库中的数据大都是细节数据，而存储在数据仓库中的数据分为两部分:概要数据和细节数据。概要数据往往比细节更能反映数据的价值，更能反映发展趋势。因此对于非授权用户来说具有非常大的吸引力，他们希望获得的数据可以是包括经济、政治、人力资源等几乎所有方面的数据。这使得数据仓库处于一种相对不安全的境地。

另一方面，通过internet，未知用户可以电子化地访问各种组织，同时由于他们是匿名的，而且可以轻易地隐藏他们的行踪，因此internet也需要有特定的安全措施。

在与 internet 结合之前，安全对于数据仓库管理员来说还是一个不可思议的问题，因为管理员可以保证使用数据仓库的人是公司的雇员。就算不是所有雇员都值得信任，大多数也是诚实可信的。对于那些不可信任的人，也会通过技术手段阻碍他们对数据仓库的使用。在 internet 推动了数据仓库的迅速发展的同时，也带来了新的安全问题：未授权用户可以更轻易地进入组织内部。在不知道闯入者意图的情况下，如果没有充足的安全措施，组织机构能够采取的措施只是希望闯入者不要做破坏活动，然而数据仓库中的丰富数据对闯入者来说却是一种强烈的诱惑。

在基于Web的数据仓库系统中，由于数据是大量集中存放，而且用户对数据仓库的应用不再局限于单一的环境，它可以为互联网上众多用户直接共享。数据仓库的分析结果往往要求能够在网上发布、浏览，让用户在线使用。这种方式在给企业的应用带来便捷的同时，也为数据仓库带来了安全隐患。概括来说，威胁数据仓库安全性的主要因素有：

1、系统认证

基于数据仓库技术的决策支持系统需要使用一层或多层安全认证过程来进行保护。一般这种认证方法是通过口令完成。涉及到口令的安全隐患很多。例如：使用本人姓名，很容易被猜到的口令或很短的口令等不正确的口令设置；虽然设置了比较安全的口令，却为记忆方便把口令记在一张很容易被人发现的字条上，在输入口令的时候被人从后面偷看；为方便数据库的口令与系统的口令相同。

2、计算机病毒

计算机病毒对于各种信息系统而言都是一个很大的安全隐患。对于数据仓库系统也不例外。新的病毒与病毒的新的变种在不断的出现，而且具有越来越大的破坏性，如CIH病毒会破坏计算机硬件；而宏病毒会随着普通文档包括WORD、EXCEL等常用文件形式进行传播。比较常见的病毒传播方式包括：公司工作人员的便携电脑安装了不正常渠道得到的游戏，连接到互联网的计算机有可能感染网络病毒；数据仓库系统的开发与维护公司与系统使用者之间有可能交叉感染。

3、电磁泄漏

一般而言，如果不采取适当的保护措施，信息系统中的电子设备包括显示器、打印机等会产生一定的电磁辐射，而这种电磁辐射可能在几百米范围内通过专用设备接受，并重现数据，造成信息泄漏。这种安全问题近年来逐渐受到研究人员的注意，原本主要在军事保密领域，考虑到数据仓库系统往往涉及到一个公司的高度商业机密，目前工业间谍的报道屡见不鲜，因而电磁泄漏需要引进高度的重视。当然，目前已经出现了专门的电磁泄漏防护设备，但是价格非常昂贵。如何进一步降低电磁泄漏防护的系统费用，是当前一个非常重要的研究课题。

4、黑客攻击

如果保护不当，大量使用互联网络的数据仓库系统很容易遭到通过网络进行的黑客攻击。不当的网络安全措施包括未安装防火墙，允许数据仓库系统的用户进行多次的登陆尝试，没有定期检查诸如BACK ORIFICE等黑客病毒、口令文件的权限包括不够、操作系统不能定期更新到最新的安全模块等。

5、网络监听

WEB环境下的数据仓库系统需要在网络环境中运行，就会有被通过网络传输线进行网络监听的可能性。只要把一个设备接到网络线上，就可能监听到网络中的传输数据，并从中窃取到重要信息，光纤比较安全，即使处在网络中的机器上也可能通过一些软件来实现这种监听工作，而与物理连接线无关。另一方面，如果监听者不仅进行监听，而且进行恶意的数据信息篡改，会带来更大的危害。

6、硬件故障、软件缺陷或错误

数据仓库系统是由许多硬件设备和软件共同构成的。其中任何一部分出现故障，或者存在缺陷和错误，都可能对整个系统的运行产生重要的影响。甚至会使整个系统瘫痪。

7、电子邮件

目前电子邮件的应用日益普及，实际上已经成为当今信息社会的一种极其重要的通讯交流手段。我们知道，电子邮件的使用也会带来一定的安全问题。使用数据仓库系统的不同部门的工作人员可能会通过电子邮件进行数据仓库系统使用或决策支持系统使用结果的交流，如果信息不加密会容易被侦听。另外，通过电子邮件，有可能绕过一般的安全保护措施，发出受到严格保护的信息数据。电子邮

件帐号的被盗用，也有可能被用来进行非法利用。

8、数据打印

通常在数据仓库系统的使用过程中，我们会打印处一些重要的数据报表或其它信息内容，以便进行更加细致的研究利用，如果保护措施不当，也会造成安全问题。例如，在一份较长时间的打印工作中途离开，就有可能被人利用窃取打印结果，而不被立即觉察。另外，普通的碎纸设备，不一定能够彻底消除信息泄漏的隐患。通过一定的恢复手段，从垃圾、废纸篓中也有可能找到很有价值的信息。一些重要的数据也有可能通过垃圾、废纸被带出公司而不被发现。

9、内置的数据库系统安全问题

大多数数据仓库过于依赖主要基于视图机制的内置安全措施，但是这种基于视图的安全措施对于数据仓库不太合适，因为如果直接进行数据卸载，可以轻松的绕过这些安全控制。另外，在数据库服务器与客户机之间的大量数据传输也可能没有加密。

10、数据库管理系统的限制

有些数据仓库系统所使用的数据库管理系统不能并发处理不同安全级别的数据。而许多机构使用同一个服务器来同时处理高度机密数据和机密数据，这有可能造成对高度机密数据的非法存取。

11、内部人员泄密

具有机密存取权限的公司雇员有可能对其合法获得的机密数据泄漏给竞争对手。公司机密财政数据也可能被某些未授权的雇员提前利用在股票市场非法牟利，而给公司造成巨大损失。

12、数据质量

数据质量问题是制约数据仓库应用的瓶颈之一。数据仓库中的数据来自不同的数据源，对这些数据的整合可能会遇到很多困难。例如，没有建立整合视图所需的公共关键字；元数据的说明不完备或者丢失；数据值相互抵触等。数据质量不高，可能会对数据仓库的完整性和可用性构成威胁。

13、其它的

包括自然灾害，比如火灾或水灾、电力中断或通信服务的中断都可能对数据仓库系统造成巨大的损失。

由于我国的数据仓库应用还处在发展的初级阶段，近年来虽然一些数据仓库产品开始关注安全问题，增加了一些安全措施，但还不够规范．基于Web的数据仓库的安全对策就更少。但我们应该看到，随着网络技术和数据仓库技术不断紧密的结合以及应用发展的需求，必须尽快地给出基于Web数据仓库的安全性策略。

5.3 WEB环境下的数据仓库采取的安全措施

5.3.1 非技术性安全措施：

1、制定安全管理制度

加强内部控制机制是数据仓库安全运行的可靠保证,只有这样,数据仓库安全工作才能做到有章可循、有的放矢。因此,应按国家的有关法律法规,根据本组织和数据仓库系统的实际情况,制定数据仓库的安全管理制度。安全管理规章要求明确数据仓库安全工作的目标,安全机构的职责,安全工作人员的权限,各部门应遵循的安全原则,安全管理工作的运行方式以及所有工作人员的职责范围,数据仓库系统受到意外损害时的应急计划,数据仓库系统安全审计的方法、内容等。

2、计算机安全立法

数据仓库系统安全问题的解决最终要依靠法制的保障。因此,有必要通过法制手段制订有关信息安全的法律规范,强制性地贯彻实施信息安全技术与安全管理等措施,保护数据仓库系统的资源不受损害。我国现有的法规政策在 3 个层次上对信息安全进行法律意义上的约束。首先,从国家宪法和其他部门法规的高度对个人、法人和其他组织的涉及国家安全的信息活动的权利和义务进行规范。其次,直接约束计算机安全和 internet 安全。最后，对信息内容、信息安全技术和信息安全产品的授权审批进行规定。其中,第一个层次上的法律主要有宪法、刑法、国家安全法和国家保密法。第二个层次主要包括《中国人民共和国计算机信息系统安全保护条例》、《中华人民共和国计算机信息网络国际互联网管理暂行规定》和《中华人民共和国计算机信息网络国际互联网安全保护管理办法》等法规。第三个层次则主要包括《电子出版物管理暂行规定》、《中国互联网络域名注册暂行管理办法》和《计算机信息系统安全专用产品检测和销售许可证管理办法》等规定。

3、信息安全的标准化

为保证数据仓库系统的安全，必须进行信息安全的标准化工作，以指导各个领

域的用户对系统正确的使用和管理。与信息安全有关的国际标准中以美国和欧洲的信息安全管理标准影响最大，在数据库领域也是最重要的标准。美国的可信计算机系统评估标准 TCSEC 手册又称橘皮书，欧洲的安全性标准简称 ITSEC,即信息技术安全性评估标准。此外，与(网络)信息安全有关的国际标准目前有两个系列：SNMP 系列和 OSI 系列。

为了保障信息系统的安全．我国的信息安全标准体系正在制定之中。《中华人民共和国计算机信息系统安全保护条例》规定，信息系统安全实行等级保护制度。安全等级保护制度的基础是信息安全标准，分为三类：安全产品标准、信息系统安全建设标准、信息系统安全管理条例。等级安全保护的基本标准有以下 12 项：

1)计算机操作系统安全评估准则

2)网络管理系统安全评估准则

3)互联网络安全管理规范

4)应用系统安全评估准则

5)计算机设备电磁辐射信息安全检测与控制标准

6)信息安全分类与系统安全保护等级划分准则

7)信息系统安全审计指南

8)信息中心安全管理规范

9)信息系统资产评估准则

10)灾变应急与恢复指南

11)信息系统风险管理指南

12)终端和用户鉴别方法

5.3.2 技术性安全措施

技术性安全措施主要从数据仓库的设计和运行两大方面进行考虑。

- 数据仓库设计时的安全措施：

1、在数据仓库设计时定义信息安全

虽然许多著作中都一致提到了在后期进行管理支持的必要，以及数据仓库对于重要商业活动的应用性，其实即使在系统运行之前，其重要性也是相当明确的，应该在设计中尽量考虑与有效性、完整性及保密性相关的内容。同时，计划中应该包含对未来业务伸缩性及增长的考虑。在用户需要得到数据仓库中的信息时，

数据仓库必须是可用的；其完整性必须得以保证，只有那些被授权的人才必须访问数据。认识到数据仓库的安全性，数据仓库同时还必须进行管理，还应当将安全问题考虑在设计方案之中。

1）数据仓库的设计必须要按照公司已经建立的信息安全政策、标准、指导方针以及相关程序来实现：在整个的设计阶段，要做出对数据仓库所预期管理功能的某些决策，如用户的数量；所需要相关网络的连接；数据建模过程和相应数据格式；实现这些设计所需要的必要资源等。这一阶段仍然存在很大的安全隐患，数据仓库的设计必须要严格遵守公司的信息安全政策，这些安全政策主要包括以下几个部分：

- 根据公司已经建立的安全政策、指导方针及其步骤对新出现的硬件和软件进行合理安装和配置；
- 对可接受的使用和相关的监控行为进行记录；
- 保证整个安全结构的一贯性；

2）数据仓库数据访问权限是为数据仓库用户群定义的

3）数据仓库中的数据内容和数据大小是在数据仓库设计中被定义和实现的

4）定义数据敏感性并且要考验其是否与其他适当的访问控制方法相互关联

5）数据的完整性和数据的推理要求是被事先定义的，同时也是与相应的访问控制相关的

6）需要定义操作系统、应用程序和通信的安全性

7）硬件配置和备份的计划必须要包含在数据仓库的设计当中

8）软件分配、配置和使用的计划必须包含在数据仓库的设计当中

9）操作和灾难恢复的连续性计划必须要包含在数据仓库的设计当中

10）对由于扩展网络连接造成结构化网络工作影响的常规评价所做出的计划必须要包含在数据仓库的设计中

2、数据仓库的安全性检查

1）要根据公司已经制定的安全政策来监控数据仓库技术的获得方式以及这些技术的安装过程

2）为了安全方面的要求 ，必须要对数据库组件的创建/产生过程进行检验

3）回顾数据仓库源数据的获得

4）对测试进行回顾

- 数据仓库运行时的安全措施：

数据仓库是一个非常复杂的系统，它的安全涉及到与其相关的数据库、网络、操作系统等的安全问题。一些通常的安全措施在数据仓库环境下仍然适用，如身份认证、访问控制、防火墙、数据加密、审计与入侵检测、系统备份和基于应用程序的安全等。很显然，上述技术中的任何一个单独使用都不能对数据仓库的所有安全问题提供一个解决的方案。因此，建立一个安全的数据仓库，这些技术必须要结合起来使用。不断增长的XML及相关安全技术的使用给我们一种提示，能否使用XML安全技术来给数据仓库提供安全。基于这种想法，在下文中提出了一个基于XML的WEB环境下的数据仓库安全模型，该模型中包含了一些常用的安全技术。

第六章 基于 XML 的 WEB 环境下的数据仓库安全模型

6.1 XML在安全方面的优势

为了更充分的理解XML在安全方面的优点，我们对比传统的网络安全方法，比如最常用于网上加密认证的是SSL和TLS协议。安全套接字层(SSL)是由Netscape公司开发的用来向因特网会话提供安全性和保密性的握手协议。它支持服务器和客户机认证,并且被设计成协商加密密钥以及在交换任何数据之前认证服务器。它使用加密、认证和MAC来维护传输信道的完整性。

虽然SSL可用于HTTP,也可以用于FTP或其它相关协议。它在传输层运行并且是独立于应用程序的,因此象FTP或HTTP之类的相关协议可以放在该层之上,使用初始握手来对服务器进行认证。在这一过程中,服务器把证书提交到客户机并指定要使用的首选密码，然后,客户机生成在即将进行的会话期间使用的秘钥,然后将它提交给服务器,并相应地用服务器的公钥对它加密。服务器使用其私钥解密消息,恢复秘钥,然后通过向客户机发送一条使用该秘钥加密的消息来向客户机认证自己。使用这一达成协议的秘钥对加密的数据进行进一步的交换。

可以用第二阶段（可选）来进一步增加安全性。这里，服务器发送一个质询，客户机对此作出响应，向服务器返回该质询的数字签名和客户机的公钥证书。质询阶段通常是使用带有用于消息摘要的MD5的RSA执行的。也可以使用各种对成密码，包括DES、三重DES、IDEA、RC2和RC4。使用的公钥证书符合X.509标准。

传输层安全性(TLS)协议是IETF标准草案,它基于SSL并与之相似，它的主要目标是在两个正在通信的应用程序之间提供保密性和数据完整性。电子文档在通过网络传输时,应保证以下四点:

1、机密性：其他任何人都不能访问或复制该数据。

2、完整性：该数据从发送者到达接收者的过程中未被更改。

3、身份验证：该文档确实发自所指的发送者。

4、不可抵赖性：发送者不可否认该文档的确由其发送,也不能否认文档的内容。

传统的SSL协议或TLS协议只能满足前三种，它们不能实现不可抵赖性。不可抵赖性是由XML安全组件所提供的一项功能，并且XML加密可以同时提供前面的三种要求。

另外SSL等协议属于点对点协议，如果要发给多个接收方就建立多个到接收方的安全链接，这会产生很多冗余操作。XML可以避免这点，它是一对多的开放性的交流方式。

最后也是最重要的一点就是XML加密不同于其他传统方法，它可以根据不同要求对电子文档的不同部分进行加密，而传统方法是你要么选择不加密文件，要么必须加密文件的全部内容。

6.2 WEB 环境下的数据仓库与 XML

WEB环境下的数据仓库能够把政府或企业中的分散的原始操作数据和各种来自外部的数据汇集成一个单一的数据集合，并且通过WEB传送系统提供给各种不同用户（在图6.1中给出了WEB环境下数据仓库系统的典型结构）。这时，如何将来自不同应用系统的分散的数据集合汇集成一个比较合理的数据集就成为比较迫切的问题。

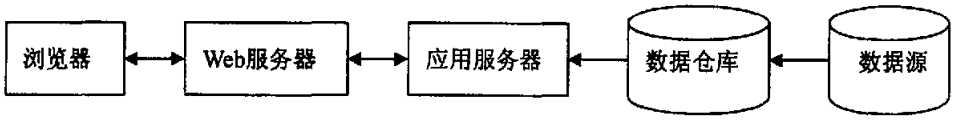


图6.1 Web环境下数据仓库系统的典型结构

不同应用系统的信息集成技术已成为近年来的研究热点之一，人们也相继提出了各种各样的解决方案。但由于应用环境不断变化，特别是 XML 的推出，新的需求问题不断出现，迫使人们不断探索新的集成方法和技术。为适合 WEB 发展的需要特别是 WEB 环境下数据仓库的要求，以 XML 作为中间层的数据描述工具和数据转换工具可以提供一种比较良好的解决方案。

将 XML 应用于数据仓库具有如下的优势：

- 容易实现数据在 WEB 上的发布，XML 数据可以不作任何修改就和 HTML 一样在网络中传输。
- 有利于数据集成，XML 可以解决异构数据源之间的兼容问题。
- 可以使用丰富的方式显示数据，表现形式多样。

- 支持本地数据处理，客户接收到数据后可以根据自己的需要解析数据，并作进一步编辑处理，减少网络流量，有利于信息共享。
- 可以实现数据的独立更新，使用XML后，一部分数据变化后，不需修改全部数据，也不影响数据表现形式。

正是因为XML的优越性，越来越多的公司通过网络用XML来传输结构化的数据，文档的安全问题也越来越重要。XML的优势来自于它的语义和结构的灵活性和可扩展性。但是正是这些优点引入了一些重要的安全问题。W3C (the World Wide Web Consortium)、IETF(the Internet Engineering Task Force)和其他几个团体参与了XML安全标准的开发工作，相关的几个重要标准有XML Encryption, XML Signature, XKMS (XML KeyManagement Specification), XACL (eXtensibleAccess Control Language)和SAML(Security Assertion Markup Language)等。XML加密保证了XML文档的内容保密性,XML签名保证了XML文档的数据完整性和不可否认，SAML和XACL共同来完成对合法用户进行授权并进行访问控制。鉴于XML安全的研究的不断深入和其在基于WEB数据仓库系统中所发挥的重大作用，本文提出一个基于XML的WEB数据仓库安全模型。

6.3 基于XML的WEB环境下数据仓库安全模型

根据前面几章内容的介绍，要想保证WEB环境下的数据仓库的安全，一般需要包含以下安全技术：身份认证、访问控制、数据加密、审计等。我们设计了一个基于XML的WEB环境下的数据仓库安全模型，拟采用以下安全技术：

身份认证－采用用户名、密码和XML签名进行双重认证。

访问控制－采用传统防火墙对内网进行保护，在传统防火墙内采用XML防火墙对SOAP消息进行检测，另外，采用基于角色的XML访问控制策略（RBAC)。

信息加密－采用XML加密技术对数据仓库中的XML数据进行加密。

审计跟踪－采用日志文件来记录所有活动。

在上面提及的安全技术中，主要使用了XML签名、XML加密、XML防火墙、XKMS、SOAP（用来进行消息的传递）等与XML相关的安全技术对WEB环境下的数据仓库的安全提供保障。图6.2给出一个框架模型：

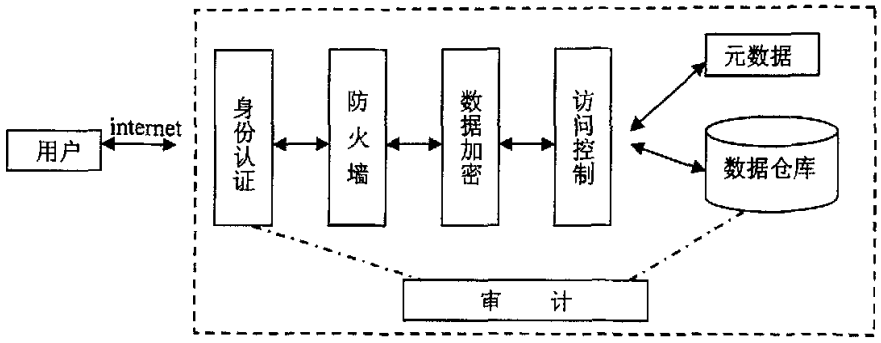


图 6.2 安全模型框架

6.3.1 安全模型的组成部分

根据上面的框架模型和具体实现的技术，我们构建了实际的安全模型。模型如图 6.3，该模型包含以下组成部分：

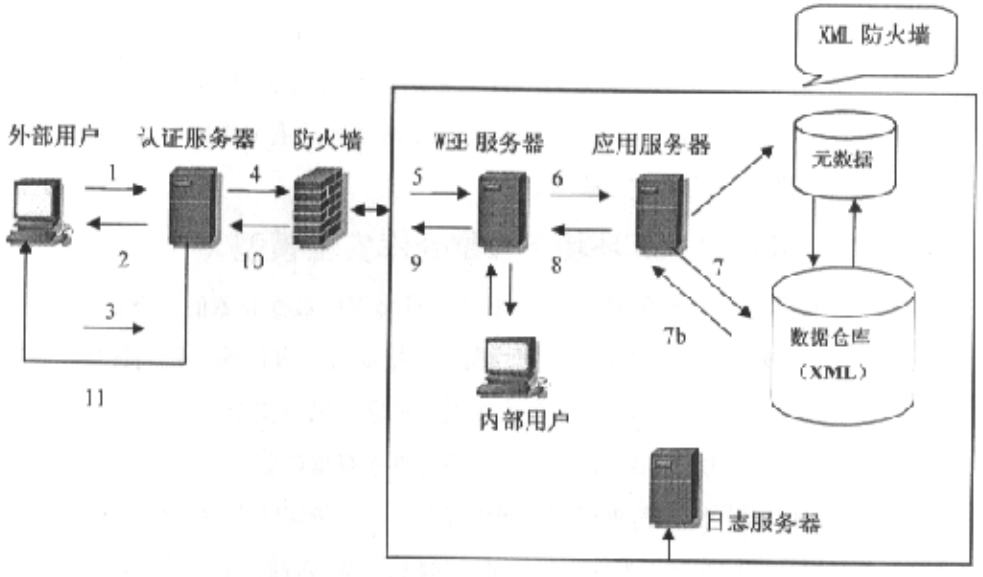


图 6.3 Web 环境下数据仓库安全模型

1）认证服务器

认证服务器用来进行用户的身份认证。这是安全模型的第一部分，在这里外部用户首次进入，发出操纵数据的请求并同认证服务器发生联系。

2）防火墙

防火墙是一个用来在网络间实现安全策略的设备。防火墙可能会含有多个网络接口，它被用来在不可信的外部网络和可信的内部网络之间建立一个边界，围绕

着数据仓库和它的核心组件。认证服务器没有放置在防火墙内。这是因为如果允许一个不怀好意的用户同防火墙内的认证服务器发生交互，可能会危及到防火墙自身的完整性。

3）XML防火墙

在传统防火墙内再使用XML防火墙。XML防火墙是一种应用防火墙，工作在应用层(applicaion-level)。它对请求Web服务的SOAP消息进行深度解析，分析其中的安全信息(包括判断消息的来源、请求者的身份和权限、SOAP消息的机密性和完整性、欲访问的Web service及其方法的安全等级等),运用设置的安全策略、访问列表和知识库对之进行验证和检测，拒绝非法的、不安全的访问，为Web service应用提供灵活的、丰富的安全解决方案。XML防火墙可以以一种应用程序的方式也可以通过WEB服务器上的服务器端软件来实现。有很多公司提供XML防火墙产品，例如DataPower Technology的XS40 XML Firewall。

4）WEB服务器

这是数据仓库安全模型的一个至关重要的部分。当用户通过了认证服务器的认证后，认证服务器将请求传递给WEB服务器，WEB服务器将会根据用户的角色来检查用户是否有权限访问这个数据。如果可以，它会送出一个请求到应用服务器，否则将根据用户权限，返回对应的错误信息，并同请求一起送回给认证服务器。

5）应用服务器

应用服务器根据WEB服务器传递过来的合法的请求进行分析和数据抽取，如果是业务数据，应用服务器就会向数据仓库发出SQL查询语句；如是元数据，则需要利用特定的API访问元数据管理系统。毫无疑问被抽取的数据是以加密的形式交给WEB服务器的。

6）数据仓库

数据仓库是从多个数据源抽取来的数据的集合。数据分为不同的分区使用不同的密钥对每个分区进行加密。数据以加密的形式传递给用户，用户根据他的角色和权限层次分配了他将要访问的分区的私钥。这些加密的数据被放置在XML格式中的一个标记之下，它会被送给用户。

7）用户

用户包括服务商、合伙人、客户等使用数据仓库的人员。每个用户被分配了：

- 用户名和密码：用来和认证服务器进行初始交互时进行鉴别。
- XML 签名：用来到认证服务器进行身份认证
- 对每个要访问的数据仓库的分区的私钥：当用户想将数据存储回数据仓库时（取决于他是否有这个权限），这些私钥用来加密数据，也用来解密从数据仓库抽取的数据。

8）日志服务器

日志服务器记载所有发生过的操作等信息。

9）元数据

它是数据仓库的核心，用于存储数据模型和定义数据结构、转换规划、仓库结构、控制信息等。通常元数据是独立于数据仓库单独存储，它有自己的数据模型和访问方式。元数据管理系统应提供询问元数据的 API。

6.3.2 安全模型涉及到的技术问题

1、从数据仓库建立时就进行考虑，将数据仓库中的数据转换为 XML 格式

XML 是互联网联合组织(W3C)于 1998 年 2 月设计的一组规范。XML 是 SGML 的简化子集,它是为 Web 应用设计的,是针对 HTML 和 Internet 设计的，标准的、可扩展的、通用的数据格式。XML 主要包括以下几个方面内容:DTD(Document Type Definition)文档类型定义,它规定了 XML 文件的逻辑结构,定义了 XML 文件中的元素、文件的属性以及元素与元素之间的关系，它可以帮助 XML 的分析程序校验 XML 文件标志的合法性；XSL(eXtensible Stylesheet Language)可扩展样式语言,适用于规定 XML 文档样式的语言,它能在客户端使 Web 浏览器改变文档的表示法,从而不需要再与服务器进行交换；XLL(Extensible Link Language)可扩展链接语言,将进一步扩展目前Web已有的简单链接。XML除了保留HTML的优点外还具有比HTML更为优越的特点:①XML 文档最显著的特点是信息的描述与信息的处理是分开的；②XML 是一种跨平台的语言；③规范简单；④很强的开放性和可扩展性；⑤自描述性。

由于数据仓库中的数据来源于多种数据源,可以是大型关系数据库（如 Oracle ,Sybase 等)、面向对象数据库(如 Objectstore)、桌面数据库(Access, Foxbase 等)、文件系统(如 Excel, Word 等)、互联网上的数据(如 Web, XML, HTML,E-mail 等),所以在数据加载到数据仓库之前,必须完成对数据的抽取、转换、

清洗、装载过程,这是构建数据仓库的重要一环。

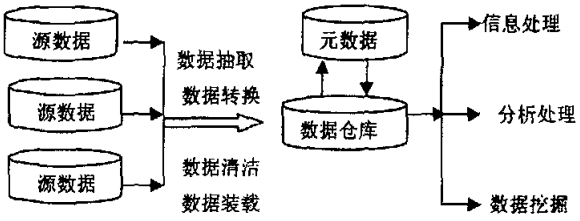


图 6.4 数据仓库的一般解决方案

文献[38]中给出了一种设计方案。设计了一个模块,负责将异构的数据源的数据转化为 XML 格式。我们将其应用在我们的数据仓库模型中,保证进入数据仓库中的数据都是 XML 格式的。

在该体系中,源数据直接进入一个 XML 格式转换器。XML 格式转换器的思想就是通过统一访问接口和不同访问实现异构数据源互连,数据源的异构性从而被屏蔽,它免除了应用开发者需熟悉各种数据源的麻烦,还可以改善应用的可移植性。这个转换器是由 XML 格式分析模块、XML 格式转换模块、XML 格式生成模块构成的。其中 XML 格式分析模块是对进入的数据进行分析,判断进入的数据格式(如 COBOL 程序、MVS 作业控制语言(JCL)、UNIX 脚本和 SQL 语句等)。XML 格式转换模块则用来把其他的数据格式转换为 XML 格式,即用 XML 格式对数据进行封装,当然我们在 XML 格式转换模块中存储了相应的格式转换程序,并在这个模块中加入了智能搜索引擎机制,使其能够自动地进行格式匹配和格式转换,这个模块是整个转换器的中枢。XML 格式生成模块即把格式转换结果进行整理,这是因为从数据库中生成的 XML 文档是一种规范格式,但在不同的应用中需要 XML 文档的不同表现形式。经过 XML 格式转换器处理过的数据,具有统一的格式,这样就大大简化了以后的 ETL 操作。

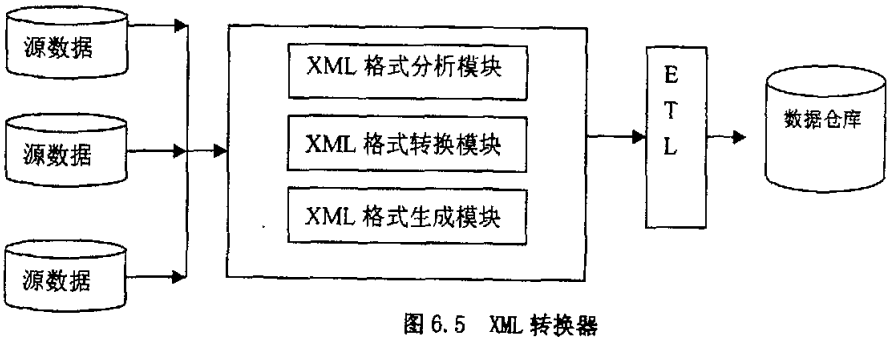


图 6.5 XML 转换器

2、基于 XML 的角色访问控制（RBAC）

在文献[39]中，给出了基于 XML 的角色访问控制，我们把它使用在我们的数据仓库安全模型中，结合其他技术共同完成数据仓库中 XML 数据的访问控制。

（1）基于角色的访问控制的相关术语

权限－用户在系统中进行任何一个操作，对信息对象的任何一种访问都会受到系统的限制，用户对特定的信息对象进行特定的操作的许可称为权限。

角色－角色与用户的权限相联系，为权限的集合。

授权－授予用户访问某种信息对象某种访问操作的权限。

继承－如果角色 A 是角色 B 的继承者，则角色享有角色 A 的所有权限。角色之间的继承关系呈现树状结构，称为角色树。

SSD(Static Separation of Duties)－角色 A 与角色 B 为 SSD 关系，当且仅当角色 A、角色 B 不能同时授予同一个用户。

DSD(Dynamic Separation of Duties)－角色 A 与角色 B 为 DSD 关系，当且仅当角色 A、角色 B 虽然可以同时授予同一个用户，但两角色中的权限不能同时执行。

（2）用户、角色、权限

用户与角色相联系，用户和角色是多对多的关系一个用户可以是一个或多个角色的成员；一个角色的成员可以是一个或多个用户。角色与权限相联系，角色与权限也是多对多的关系。一个角色可以拥有一种或多种权限；一种权限可以授予一个或多个角色。用户、角色和权限的关系如图 6.6 所示。

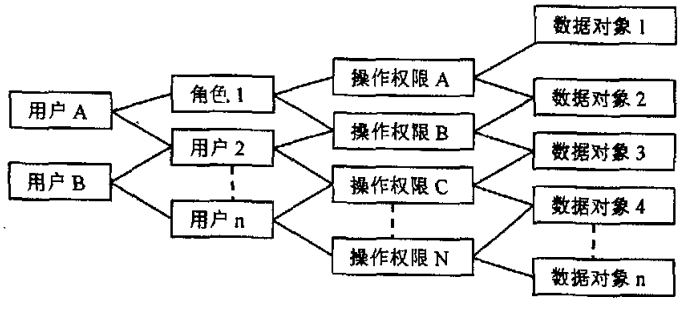


图 6.6 用户、角色、权限的关系

（3）RBAC 机制的实现

用户拥有惟一的标志号 ID 和密码 Password，用户被赋予角色的成员身份。

当用户需要对某信息对象(Resource)执行相应操作(Action)时,RBAC 执行其功能。它根据此用户的角色,查看此角色是否有此用户所要求的权限,若有则允许用户执行权限,否则禁止。RBAC 实现机制如图 6.7 所示。

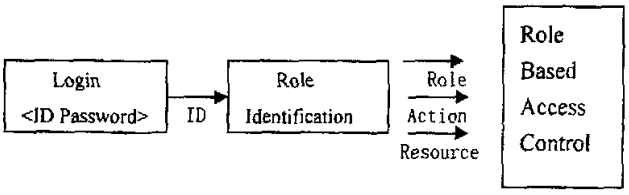


图 6.7 RBAC 机制

(4) 角色关系和角色权限的 XML 表示示例

下面给出了一个记录银行角色间关系的 XML 文件 role.xml 和记录银行角色权限的 XML 文件 privilege.xml。

role.xml

```
<?xml version=" 1.0" ?>
<roles>
<rolename=" Visitor" >
<membership>
<rolename=" Accountholder" dsd=" Teller" />
<rolename=" Invitedguset" />
<rolename=" Employ" >
<membership>
<rolename=" Branchmanager" />
<rolename=" Internalauditor" />
<rolename=" Teller" dsd=" Accountholder" />
<rolename=" Accountrep" >
<membership>
<rolename=" Branchmanager" />
</membership>
</role>
</membership>
```

```
</role>
</membership>
</role>
</roles>
```

privilege.xml

```
<?xmlversion=" 1.0" ?>
<privilegeinfo>
<rolename=" Visitor" >
<itemresource=" tname" action=" select" />
<itemresource=" tsalary" action=" select" />
</role>
<rolename=" Teller" >
<itemresource=" tname" action=" update" />
<itemresource=" tsalary" action=" update" />
<itemresource=" http://mouse/resource/notice.txt " action=" open"
/>
</role>
</privilegeinfo>
```

(5) 在安全模型中的实现

用户在首次访问数据仓库之前，需要通过WEB注册、EMAIL获取等方式取得用户名和口令，同时系统管理员为其定义角色及相应的权限。

角色权限的认证主要在模型中的WEB服务器中完成。如果用户当前的角色具有它所要求的权限，由WEB服务器将请求发送给应用服务器完成数据的抽取工作。

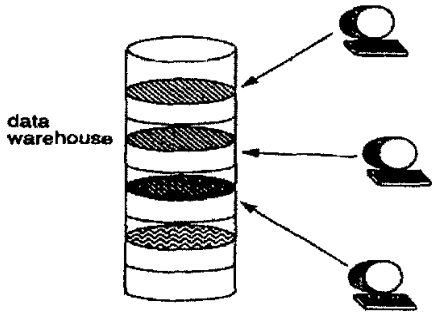


图 6.8 用户根据角色访问数据仓库的不同部分

3、传统防火墙和 XML 防火墙相结合

防火墙是围绕着数据仓库部署的，它能对安全的区域提供了一个更加安全的屏障，以防止非法入侵。传统的防火墙还存在着一些不足之处，因为它们仅能在信息包层提供过滤，而不能检查消息的内容。传统的防火墙通过几个不同的方式来阻碍通过 Internet 进行的通信，以对网络的边界提供保护。除了端口 80（用于 HTTP 通信），端口 443（HTTPS 通信）和端口 25(用于 EMAIL 通信)以外，它们会阻碍所有的 TCP 端口。一个明智的选择是在网络中使用 XML 服务，允许使用 XML 防火墙。XML 防火墙的典型工作是检查 SOAP 信息头，信息头可能包含一些详细的信息，放在那里供防火墙来检测。如果这样，防火墙可以针对这些信息采取行动。即使头部没有这些信息，XML 防火墙仍能基于头部所包含的内容采取行动。头部信息，举例来说，可能包含了信息收件人的名称，包含了全部信息的安全内容或者是关于信息传输所经过的中间层的内容。

另外，XML 防火墙能够查看信息体部分并且根据标签层来检查它，它能告知一个信息是否是一个授权的，或者来自一个经过授权的收件人。然后基于此采取相应的行动。举例来说，它可以阻碍通信，也可以把它们送至一个安全的环境以进行进一步的检查，或者允许它通过。XML 防火墙还提供其他的保护方法，它能理解关于操作的元数据，也能一样去理解关于 WEB 服务请求的元数据。它们能从提出请求的用户那里收集信息。比如理解用户在当前的请求中扮演了什么样的角色。例如，XML 防火墙能提供认证、解密、实时监控和报告。所以当用户传送过来一个请求时，XML 防火墙也能查看它的内容。

在我们的安全模型中，将传统防火墙和XML防火墙整合起来协同工作，在网络

防火墙所保护的网络区域内部，再使用XML防火墙保护Web Services服务。

4、对数据仓库数据进行加密

数据仓库是一个数据的集合，它可能包含了TB级的数据，并不是所有的用户都要访问全部的数据。为了实现访问，用户被分成不同的角色，并且根据它们的需求给予不同的权限和许可。如果要进一步增加安全，数据应该被分段，每一个部分都应该使用独立的密钥进行加密。然后根据各自的权限这些密钥被分发给不同的用户。对称加密很显然是这项工作的一个参与者。但单个密钥非常不安全，我们还可以使用非对称加密技术。即使用公钥对数据进行加密，发给用户的私钥用来解密。这样的话，如果一个不怀好意的用户妄图插手数据，他需要得到一个私钥来解密数据。不管使用两种加密方法中任何一种，在任何被允许的处理之前首先进行用户的认证都是非常明智的。

在我们的安全模型中，采用了 XML 加密技术对数据进行加密。XML 加密的优越性在前面已经提过，它可以支持对 XML 文档的一些特定部分进行加密。对于同一个文档中的不同部分用不同的密钥进行加密，就可以把同一个 XML 文件发给不同的接收者，而接收者只能看见和他相关的部分。

在加密技术的选择上，我们综合采用了对称加密和非对称加密技术。具体实现可以采用对称密钥算法（比如三重 DES）来加密数据仓库中的数据,然后使用用户的公钥来加密这一对称密钥。将这两个数据项都传给预计的接受者。接受者收到加密过的两个数据项时，首先使用自己的私钥来解密出对称密钥，然后使用合适的对称算法来解密数据。这两种方法相结合，有效地解决了对称密钥传递的安全性问题和 RSA 算法效率低的问题。

5、使用 XML 签名来进行用户的身份认证

W3C 将 XML 数字签名解释为：定义一种与 XML 语法兼容的数字签名语法描述规范，描述数字签名本身和签名的生成与验证过程。作为一个安全有效的数字签名方案，该规范提供了数字签名的完整性、签名确认性和不可抵赖性。为了适应不同的文件系统和剖析器，XML 签名非常依赖“规范化”的概念，以便签名能够在 XML 文档所碰到的各种环境中起作用。在目前的规范中规定：消息摘要使用 SHA—1 算法，消息鉴别编码使用 HMAC 算法，数字签名使用 DSA 和 PKCS1 算法；在安全模式上，可以使用基于对称密钥和基于公开密钥两种体制。

在我们的模型中，使用了 XML 签名来进行用户的身份认证。具体签名的实施，在文献[33]中给出了数字签名应用原型 AppSign。当然，我们也可以采用一些已有的工具包。IBM 的 alphaWorks 提供了 IBM 的安全组件 XML Security Suite[40]，该安全组件中有一个自动生成 XML 数字签名的工具，使用它即可给 XML 文件签名。此外，微软也发展了一套基于数字证书的 XML signed 认证方式，使用微软提供的 XMLsign.exe 工具，也可以给 XML 文件数字签名。我们在本章的最后，给出了使用 XML Security Suite 安全组件来生成数字签名的步骤。

6、使用 SOAP 来进行消息传输

SOAP 为应用程序提供了一种通过 HTTP 或 SMTP 协议在网络上发送基于 XML 消息的方法。在我们的模型中，用户和数据仓库之间可以使用消息进行通信，采用了 SOAP 协议在 HTTP 协议的基础上进行通信。SOAP 消息从根本上说是从发送者到接受者之间进行单向传送的，但 SOAP 消息通常和请求/响应模式进行混合。

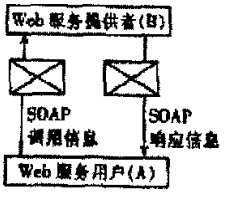


图 6.9 SOAP 消息的请求/响应模式

在我们的模型中，从认证服务器发送一条 SOAP 消息，在 SOAP 消息头中，包含了发送者的信息，在 SOAP 体中包含了用户的请求信息。XML 防火墙检查 SOAP 消息的头文件以验证消息是否由认证服务器发出。

当用户所需要的信息从数据仓库中取出后，WEB 服务器也会建立一个 SOAP 消息，将信息放在 SOAP 体中，送给 XML 防火墙。

为了保证 SOAP 消息自身的安全，还使用了 XML 签名来保证消息的完整性。每个用户通过他们的签名进行鉴别。SOAP 安全扩展通过定义<SOAP-SEC:signature>这个 SOAP 头部入口提供了一种标准的方式使用 XML 签名来签署 SOAP1.1 信息。另外，SOAP 安全扩展重用了两个存在的 SOAP 头，它们是’ACTOR’头，用来指定头部元素中的收件人，另一个是“mustunderstand”头，用来确认被附着的 XML 数字签名。为了免受所谓重放攻击，必须结合一定的方法来保证消息的唯一性，如时间戳或 nonces 等。签名日期和时间都附加在消息上，并与消息一起签名。添加

这些信息可以在其中加入扩展元素来实现。这样，消息的接受方在收到消息时可以对消息的唯一性表示进行验证。如果发生冲突或者不一致，接受方可拒绝发送方请求。

7、密钥的注册和管理

密钥的注册和管理是由 XKMS 来负责的。服务器和系统中注册的每个用户都被分配了一个公钥和一个私钥。

6.3.3 安全模型的工作步骤

安全模型的工作步骤如下：

1、用户在访问数据仓库之前，必须使用用户名和密码进行登陆。用户名和密码的获取有多种方式：WEB 注册、EMAIL 获取等。在获得用户名和密码的时候，系统管理员定义其角色及相应的权限，并在相应的服务器端保留该角色及权限信息。作为初始步骤，用户输入其用户名和密码同认证服务器进行交互。

2、如果口令正确，作为第二种验证方法，认证服务器将会发送给用户一个文档，用户将会使用 XML 签名进行文本的数字签名。文档的一部分由认证服务器自身进行了数字签名以确定其自己的身份。同时写一个带有日期和交互时间和用户名的日志文件。否则，服务器产生一个错误信息和日志送到日志文件中。将要送给用户的文档每次是随机产生的。

3、在检验了验证服务器的 XML 数字签名之后，用户使用自己的 XML 签名签署文件。认证服务器和用户的 XML 签名的验证及签名密钥的分发和管理均使用了 XKMS 和它的两个协议 X-KISS 和 X-KRSS。检验完后，用户将文本同请求一起送回到认证服务器。

4、认证服务器检验用户的 XML 签名和时间戳，如果发现其合法，发送一个 SOAP 消息，将请求放置在 SOAP 消息体中并和日志信息一起送往防火墙内的 WEB 服务器。

5、XML 防火墙检查 SOAP 消息头文件中的信息以验证消息是否是从认证服务器发出的，如果是的，它只允许请求部分传输到 WEB 服务器。

6、WEB 服务器根据用户的角色检查请求是否是在用户的权限范围之内，如果是的话，它将请求送到应用服务器，否则产生一个错误信息并将请求返回到防火墙。接下来防火墙将错误信息送回认证服务器。

7、 应用服务器从数据仓库抽取到所需要的信息，当然是以加密的形式。

8、 应用服务器将数据送回WEB服务器。

9、 WEB服务器建立一个SOAP信息，将用户所需的信息以加密的形式放置其中，并送给XML防火墙。

10、防火墙将数据同包含请求信息的日志一起送回认证服务器。从认证服务器送出的日志被送到日志服务器。为达到这个目的，防火墙必须要读信息并且被允许抽取所需要的信息。

11、认证服务器将用户所需的信息送给用户。

12、用户抽取到所需的数据，毫无疑问是以加密的形式存在的，用户使用他自己的私钥对数据进行解密。

6.4 安全模型的数字签名部分的实现

模型的数字签名的实现，可以采用IBM公司的XML安全组件XML Security Suite，它提供了一个自动生成数字签名的工具，可以对XML文件进行数字签名，方法如下：

1、准备工作

下载安装 Java 2 Development Kit ，文件大小 37MB ；然后下载安装 XML Security Suite ，将 /xss4j/samples 和 xss4j.jar 添加至你的 classpath 中。

2、创建自己的数字证书

在对 XML文件数字签名之前，首先应该有一个X.509数字证书，你可以用 Java 2 的 keytool 命令创建这个数字证书，方法如下：进入命令提示符下，CD命令进入X:\Program Files\j2sdk1.4.1_07\bin目录，输入以下命令

keytool -genkey -dname "CN=lhl, OU=cwk, O=zlsgs,L=liuan, S=anhui, C=china" -keypass 123456 -storepass security -alias xss4j

在以上命令中，dname称为特异名，特异名在整个因特网上都是唯一的，dname由普通名(CN)、组织单元(OU)、组织(O)、区域 (L)、州(S)和国家(C)组成；密匙库 (-storepass) 的密码为 security，此证书的私钥密码 (-keypass)为123456，而 xss4j 则是认证 (-alias) 的别名。

3、如何对内部XML资源数字签名

下面我们要用刚才生成的数字证书，对 d:/xss4j/samples/sonnet.xml 文件

进行数字签名，首先我们看看如何对内部 XML 资源数字签名？如果你要以这种方式对 sonnet.xml（莎士比亚的十四行诗）签名，签名过后，将会产生一个新文件 lh.XML，该文件内容如下所示，包含了签名信息、密钥和原来 sonnet.xml 文件的内容。

```
<?xml version="1.0" encoding="UTF-8"?>
<Signature xmlns="http://www.w3.org/2000/01/xmldsig/">
  <SignedInfo>
    <CanonicalizationMethod
Algorithm="http://www.w3.org/TR/1999/WD-xml-c14n-19991115"/>
    <SignatureMethod
Algorithm="http://www.w3.org/2000/01/xmldsig/dsa"/>
    <Reference IDREF="Res0">
      <Transforms>
        <Transform
Algorithm="http://www.w3.org/TR/1999/WD-xml-c14n-19991115"/>
      </Transforms>
      <DigestMethod
Algorithm="http://www.w3.org/2000/01/xmldsig/sha1"/>
      <DigestValue
Encoding="http://www.w3.org/2000/01/xmldsig/base64">
        WfKRfGWiAni9bY9k/sXgBFt4ge4=
      </DigestValue>
    </Reference>
  </SignedInfo>
<SignatureValue>
  MCwCFBOM62GxrrxMGm7qfBt8R+Zv4YuIAhRvQH1DkgAdtnDQIOYOg07srWhaiA= =
</SignatureValue>
<KeyInfo>
  <X509Data>
```

斜体表示实际签名的信息

用来验证签名的密钥

<X509Name>CN=Doug Tidwell, OU=developerWorks, O=IBM, L=Research Triangle Park, ST=North Carolina, C=US</X509Name>

<X509Certificate>

MIIDQTCCAv8CBDj6LR4wCwYHKoZIzjgEAwUAMIGFMQswCQYDVQQGEwJVUzEXMBUGA1UECB
MOTm9ydGggQ2Fyb2xpbmExHzAdBgNVBAcTF1J1c2VhcmNo1FRyaWFuZ2x1IFBhcmsxDDAK
BgNVBAoTA01CTTEXMBUGA1UECxMOZGV2ZWxvcGVyV29ya3MxFTATBgNVBAMTDERvdWcgVG
1kd2VsbDAeFw0wMDA0MTYyMTE0MDZaFw0wMDA3MTUyMTE0MDZaMIGFMQswCQYDVQQGEwJV
UzEXMBUGA1UECBMOTm9ydGggQ2Fyb2xpbmExHzAdBgNVBAcTF1J1c2VhcmNo1FRyaWFuZ2
x1IFBhcmsxDDAKBgNVBAoTA01CTTEXMBUGA1UECxMOZGV2ZWxvcGVyV29ya3MxFTATBgNV
BAMTDERvdWcgVG1kd2VsbDCCAbgwggEsBgcqhkjOOAQBMIIBHwKBgQD9f1OBHXUSKVLfSp
wu7OTn9hG3UjzvRADDHj+At1EmaUVdQCJR+1k9jVj6v8X1ujD2y5tVbNeBO4AdNG/yZmC3
a51QpaSfn+gEexAiwk+7qdf+t8Yb+DtX58aophUPBPuD9tPFHsMCNVQTWhaKMvZ1864rYd
cq7/IiAxmd0UgBxwIVAJdgUI8VIwvMspK5gqLrhAvwWBz1AoGBAPfhoIXWmz3ey7yrXDa4
V7151K+7+jrqgv1XTAs9B4JnUV1XjrrUWU/mcQcQgYC0SRZxI+hMKBYTt88JMozIpuE8Fn
qLVHyNK0Cjrh4rs6Z1kW6jfwv6ITVi8ftiegEk08yk8b6oUZCJqIPf4Vr1nwaSi2ZegHtV
JWQBTDv+z0kqA4GFAAKBgQD565x1w4WYYb0xHbSTS4k14xQh9b0A1awU+A30ex4q758MKy
1t4eX1QCDRKrxFr/bY3JsOGI6Nd6afT9FAn++2uK9yWcgTSmM6aiAqmbOEiybf50q7daX5
5ycr/5IZtLKJqNhJznnUvkqZQ2ce+6s1au1J8cw7yALDZJhzfF1hHzALBgcqhkjOOAQDBQ
ADLwAwLAIUZUJDYdoysr2us1eZ9p6xRwEZYnoCFC97/v+w6FIyM+KgDDAgK9LYxN/n

</X509Certificate>

</X509Data>

</KeyInfo>

<dsig:Object Id="Res0" xmlns="" xmlns:dsig="http://www.w3.org/2000/01/xmldsig/"><sonnet type="Shakespearean">

<author>

<last-name>Shakespeare</last-name>

<first-name>William</first-name>

<nationality>British</nationality>

Sonnet xml 文件的内容

```
    <year-of-birth>1564</year-of-birth>
    <year-of-death>1616</year-of-death>
  </author>
  <title>Sonnet 130</title>
  <lines>
    <line>My mistress' eyes are nothing like the sun,</line>
    <line>Coral is far more red than her lips red.</line>
    <line>If snow be white, why then her breasts are dun,</line>
    <line>If hairs be wires, black wires grow on her head.</line>
    <line>I have seen roses damasked, red and white,</line>
    <line>But no such roses see I in her cheeks.</line>
    <line>And in some perfumes is there more delight</line>
    <line>Than in the breath that from my mistress reeks.</line>
    <line>I love to hear her speak, yet well I know</line>
    <line>That music hath a far more pleasing sound.</line>
    <line>I grant I never saw a goddess go,</line>
    <line>My mistress when she walks, treads on the ground.</line>
    <line>And yet, by Heaven, I think my love as rare</line>
    <line>As any she belied with false compare.</line>
  </lines>
</sonnet></dsig:Object>
</Signature>
```

要创建这样的数字签名，你可以使用 SampleSign 应用程序，这个程序是 XML 安全组件附带的，你可以在 xss4j/samples 目录中找到之。以下是对 sonnet.XML 进行数字签名的方法：

进入命令提示符下， CD 命令进入 X:\Program Files\j2sdk1.4.1_07\bin 目录，输入以下命令（注意以下命令是单行的）

```
java SampleSign xss4j security 123456 -embXMLfile:///d:/xss4j/samples/sonnet.XML>
lh.XML
```

请注意：上面命令中的别名（ xss4j)、密匙库密码（security)、私钥密码（123456)、与以上 keytool 命令中的相同，这里用 file: URL 代替了简单的文件名，而输出内容则送至（用 >操作符）文件 lh.xml 中。执行过该命令之后，利用证书对内部 XML 资源数字签名就大功告成，结果会产生一个签名过的 XML 文件 lh.XML。

4、如何对外部 XML 资源数字签名

对外部 XML 资源签名，表明文件中的<Signature>下包含 XML 资源的 URL，而不是资源本身，即分离的数字签名。要创建这样的数字签名，应使用 -extXML 选项，方法如下：进入命令提示符下，CD 命令进入 X:\Program Files\j2sdk1.4.1_07\bin 目录，输入以下命令

java SampleSign xss4j security 123456 -extXMLfile:///d:/xss4j/samples/sonnet.XML> external-lh.XML

于是将产生一个签名过的 XML 文件 external-lh.XML，它与以上的 lh.XML 相似，不过 sonnet.xml 文件的内容，并没有被复制到 external-lh.xml 中的<Signature>下。

5、验证数字签名

XML 安全组件还提供了一个实用程序 SampleVerify，用来验证数字签名。使用它，你可以检查某个签名过的 XML 文件，以便确认被签名的资源没有被篡改，此外还能检查该签名与发送者的证书信息是否相符。

例如要验证 external-lh.xml 中的数字签名，你可以这样操作：进入命令提示符下，输入以下命令

java SampleVerify -dom < external-lh.XML

显示结果如下：

Signer: CN=lhl, OU=cwk, O=zlsgs,L=liuan, S=anhui, C=china

SignedInfo Bytes: 1069

--> Location: file:///d:/xss4j/samples/sonnet.XML

Validity: Ok

--> SignedInfo: Ok

```
--> All: Ok?
```

从以上命令的显示结果可知，该数字签名有效， 签名过的文件没有被改动。假如签名过的文件被改动了，以上命令执行后，将会显示不同的结果，表示该签名不再有效，于是改动过的 XML 文件同数字签名就不符，我们即可知道不能信任该文件。

第七章 结论与展望

7.1 结论

随着数据仓库的日益普及，安全性无疑将成为一个重要的研究和应用领域。进入数据仓库中的数据在组织结构和分布方面都发生了重大的变化，其价值已经提升，而借助现代数据分析技术，这些数据还可以衍生出新的、具有创造性的知识。这种再加工的过程意味着更大的增值。但随之而来的却是数据仓库的安全的思考：一方面，数据分析需要更好、更便利的数据访问模式；另一方面，这些数据蕴含的价值使得其拥有者要求限制数据的访问。显然这两个需求是相互矛盾的，由此产生一个新的安全问题：如何在不干扰正常数据分析的前提下，提供更加安全的数据访问？

在基于 WEB 的数据仓库系统中，由于数据是大量集中存放，而且用户对数据仓库的应用不再局限于单一的环境，它可以为互联网上众多用户直接共享。数据仓库的分析结果往往要求能够在网上发布、浏览，让用户在线使用。这种方式在给企业带来便捷的同时，也为数据仓库带来了更大的安全隐患。

WEB环境下的数据仓库的安全是一个综合广泛的概念，其技术覆盖了几乎所有的安全领域。一些通常的安全措施在数据仓库环境下仍然适用，如身份认证、访问控制、防火墙、数据加密、审计和基于应用程序的安全等。这些技术单独使用并不能解决所有的数据仓库的安全问题，只有将这些安全技术结合起来，才有可能取得较为满意的效果。XML及相关的安全技术很显然是解决数据仓库安全的一个参与者。将XML应用于数据仓库具有很多优势，因此基于XML的数据仓库的安全模型的研究具有很重要的意义。当然，在满足使用的便利性、可靠性和稳健性方面，XML安全技术还有很多路要走。但是目前，正在取得良好的进展。

7.2 下一步的工作展望

目前数据仓库获得了广泛的应用，越来越多的企业致力于数据仓库的开发和应用。数据仓库的安全特别是 WEB 环境下的数据仓库的安全将显得越来越重要。本文对实现数据仓库的安全方面作了一定的工作，今后还可以在以下几个方面开展进一步的工作：

1）将提出的数据仓库安全模型具体实现，并通过实验的方式验证它的可行性，进一步从理论上和实践上完善该模型。

2）把基于 XML 的 WEB 数据仓库安全模型应用到系统中去，保障系统的安全，使其更好的服务于企业。

参考文献

[1] （美）W.H.Inmon. 王志海等译. 数据仓库[M] 北京 机械工业出版社，2003

[2] 石丽，李坚. 数据仓库与决策支持[M] 北京 国防工业出版社，2003

[3] 仲红，谢荣传. 基于Web的数据仓库. 安徽师范大学学报，2002，25(2)

[4] 王珊. 数据仓库技术与联机分析处理[M] 北京 科学出版社，1998

[5] Kirkgoze R, Katic N, Stolba M, et al. A Security Concept for Olap. Proc of the 8th International Workshop on Database and Expert System Application(DEXA'97). IEEE Computer Press. 1997,619-626

[6] Harmon C. Safeguarding the data warehouse. Computer Fraud & Security. 1998, 6:16-19

[7] W.H.Inmon. Data Warehouse and Internet Security. http://www.inmoncif.com.1997

[8] Wareigon S. Data warehouse control and security. Association of College and University Auditors LEDGER 1997, 41(2):3-7

[9] Abelló A, Oliva M, et al. Information system architecture for secure data warehousing. Proc of the 3rd Workshop EFIS 2000.2000,33-40

[10] W.H.Inmon. Data Warehouse and Security. http://www.inmoncif.com . 2000

[11] Weippl E, Mangisengi O, Essmayr W, et al. An authorization model for data warehouse and OLAP. Proc of the Workshop on Security in Distributed Data Warehousing, in Conjunction with 2th. IEEE Symposium Reliable Distributed Systems (SRDS'2001). 2001

[12] Katie N, Quirchmayr G, Schiefer J, et al. A prototype model for data warehouse security based on metadata. Proc of the 9th International Workshop on Database and Expert Systems Applications. 1998, 300-308

[13] A. Perkins, Developing a Data Warehouse, the. Enterprise Engineering Approach, Visible Systems Corporation. 1995-96

[14] Rosenthal A, Sciore E. View security as the basis for data warehouse security. Proc of theInternational Workshop on Design and Management of Data

Warehouse(DMDW'2000). 2000

[15] Piatini Mario, Rodero J A. Auditing data warehouse security. Proc of IEEE 33rd Annual 1999 International Carnahan Conference on Security Technology. 1999, 255-261

[16] Priebe T, Pemul G. Towards OLAP security design-survey and research issues. Proc of the 3rd ACM International Workshop on Data Warehousing and OLAP(DOLAP 2000), 2000, 33-40

[17] Priebe T, Pemul G. A pragmatic approach to conceptual Modeling of OLAP security. Proc of 20th International Conference on Conceptual Modeling(ER 2001). Lecture Notes in Computer Science. 2001, 2224:331—24

[18] 唐蕾，徐洁磐. 基于WEB的数据仓库安全模型分析与探讨. 计算机应用研究，2004.12

[19] 李海泉，李健. 计算机系统安全技术[M] 北京 人民邮电出版社，2001

[20] 刘启原，刘怡. 数据库与信息系统的安全[M] 北京 科学出版社，2000

[21] 张世永主编. 网络安全原理与应用[M] 北京 科学出版社，2003

[22] (美)William Stallings. 杨明等译.密码编码学与网络安全：原理与实践（第二版）[M] 北京 电子工业出版社，2001

[23] (美)Harold F.Tipton,Micki Krause.王顺满等译. 信息安全管理手册（卷Ⅱ）[M] 北京 电子工业出版社，2004

[24] (美)Elliote Rusty Harold. 杜大鹏，李善茂，傅烨等译.XML实用大全[M]. 北京 中国水利水电出版社，2000

[25] (美)Blake Dournaee. 周永彬等译. XML安全基础. 北京 清华大学出版社，2003

[26] (美)Ben Galbraith,Whitney Hankison. 吴旭超，王黎译. Web服务安全性高级编程 北京 清华大学出版社，2003

[27] World Wide Web Consortium .Extensible Markup Language(XML)[EB/OL] http://www.w3.org/XML/

[28] XML Encryption Syntax and Processing.W3C Recommendation 10 December 2002 http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/

[29] XML-Signature Syntax and Processing W3C Recommendation 12 February 2002 http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/

[30] XML Key Management Specification (XKMS 2.0). W3C Candidate Recommendation 5 April 2004. http://www.w3.org/TR/2004/CR-xkms2-20040405/

[31] XML Access Control Language(XACL). http://www.trl.ibm.com/projects/xml/xacl/xacl-spec.html

[32] Simple Object Access Control. W3C Recommendation 24 June 2003 http://www.w3.org/TR/soap12

[33] 谢铉洋，谢荣传. XML 数字签名. 计算机应用研究, 2002.7

[34] 李波，易虎，李光耀. 基于 SOAP 扩展加强 Web 服务安全的方法. 微型机与应用,2004.6

[35] 曾牛，程胜利. XML 防火墙在 Web Services 安全中的应用研究. 武汉大学学报（理学版），2004.10

[36] 董金新,张虹. 证券系统数据仓库的安全机制设计. 微型机与应用，2003.11

[37] 周晓光,朱蓉,聂规划. 数据仓库的安全与对策研究. 武汉理工大学学报，2002,24(3)

[38] 仇丽青，王敏，赵庆祯. 面向 Web 的数据仓库体系设计. 计算机应用研究. 2004.9

[39] 耿晖，王海波. 基于 XML 的角色访问控制（RBAC）. 计算机应用研究. 2002.12

[40] Doug Tidwell. XML 安全组件：增强电子商务的安全性 [EB/OL]. http://www-128.ibm.com/developerworks/cn/xml/xmlsecuritysuite/index.html

[41] IBM developerWorks 中国网站编辑组. XML 安全专题 [EB/OL]. http://www-128.ibm.com/developerworks/cn/xml/theme/x-security.html

致 谢

首先深深感谢我的导师谢荣传副教授!在我论文的写作过程中，自始至终得到谢老师的亲切关怀和悉心指导。谢老师渊博的学识、一丝不苟的工作作风都给我的学习和研究以莫大的启发。他严谨的学风、豁达的胸襟和诲人不倦的精神令我终生难忘。

感谢生我养我的父母双亲，他们对我无私的关爱和付出为我营造出一个良好的生活和学习环境。

感谢我的妻子宣宾博士，她在我的学习上和生活上给了我莫大的鼓励和帮助。

感谢所有关心和帮助过我的人。

攻读学位期间发表的论文

[1] 袁学松，宣宾. 高校招生远程录取的有关技术问题探讨.
皖西学院学报. 2005, 21(2):105-108

[2] 袁学松.数据仓库安全技术研究. 安徽电子信息职业技术学院学报.(已录用)

Web环境下数据仓库安全技术研究

作者：袁学松
学位授予单位：安徽大学

参考文献(42条)

1. 参考文献

2. W H Inmon. 王志海. 林友芳 数据仓库 2003

3. 石丽. 李坚 数据仓库与决策支持 2003

4. 仲红. 谢荣传 基于Web的数据仓库[期刊论文]-安徽师范大学学报(自然科学版) 2002(2)

5. 王珊 数据仓库技术与联机分析处理 1998

6. Kirkgoze R. Katic N. Stolba M A Security Concept for Olap 1997

7. Harmon C Safeguarding the data warehouse 1998

8. W H Inmon Data Warehouseand Internet Security 1997

9. Wareigon S Data warehouse control and security 1997(02)

10. Abelló A. Oliva M Information system architecture for secure data warehousing 2000

11. W H Inmon Data Warehouse and Security 2000

12. Weippl E. Mangisengi O. Essmayr W An authorization model for data warehouse and OLAP 2001

13. Katie N. Quirchmayr G. Schiefer J A prototype model for data warehouse security based on metadata 1998

14. A Perkins Developing a Data Warehouse, the. Enterprise Engineering Approach 1995

15. Rosenthal A. Sciore E View security as the basis for data warehouse security 2000

16. Piatini Mario. Rodero J A Auditing data warehouse security 1999

17. Priebe T. Pemul G Towards OLAP security design-survey and research issues 2000

18. Priebe T. Pemul G A pragmatic approach to conceptual Modeling of OLAPsecurity 2001

19. 唐蕾. 徐洁磐 基于Web的数据仓库安全模型分析与探讨[期刊论文]-计算机应用研究 2004(12)

20. 李海泉. 李健 计算机系统安全技术 2001

21. 刘启原. 刘怡 数据库与信息系统的安全 2000

22. 张世永 网络安全原理与应用 2003

23. William Stallings. 杨明. 齐望东 密码编码学与网络安全:原理与实践 2001

24. Harold F Tipton. Micki Krause. 王顺满 信息安全管理手册 2004

25. Elliote Rusty Harold. 杜大鹏. 李善茂. 傅烨 XML实用大全 2000

26. Blake Dournaee. 周永彬. 贺也平. 刘娟 XML安全基础 2003

27. Ben Galbraith. Whitney Hankison. 吴旭超. 王黎 Web服务安全性高级编程 2003

28. World Wide Web Consortium. Extensible Markup Language(XML)

29. XML Encryption Syntax and Processing. W3C Recommendation 2002

30. XML-Signature Syntax and Processing W3C Recommendation 2002

31. XML Key Management Specification (XKMS 2.0). W3C Candidate Recommendation 2004

32. XML Access Control Language(XACL)

33. Simple Object Access Control. W3C Recommendation 24 June 2003 2002

34. 谢铉洋. 谢荣传 XML数字签名[期刊论文]-计算机应用研究 2002(7)

35. 李波. 易虎. 李光耀 基于SOAP扩展加强Web服务安全的方法[期刊论文]-微型机与应用 2004(6)

36. 曾牛. 程胜利 XML防火墙在Web Services安全中的应用研究 2004(10)

37. 董金新. 张虹 证券系统数据仓库的安全机制设计[期刊论文]-微型机与应用 2003(11)

38. 周晓光. 朱蓉. 聂规划 数据仓库的安全与对策研究 2002(03)

39. 仇丽青. 王敏. 赵庆祯 面向Web的数据仓库体系设计[期刊论文]-计算机应用研究 2004(9)

40. 耿晖. 王海波 基于XML的角色访问控制(RBAC)[期刊论文]-计算机应用研究 2002(12)

41. Doug Tidwe XML安全组件:增强电子商务的安全性

42. IBM developerWorks. 中国网站编辑组 XML安全专题

相似文献(10条)

1. 学位论文 张开松 基于Web技术的数据仓库研究与设计 2005

随着数据仓库和web技术的迅猛发展,人们对数据仓库和web技术的研究越来越广泛,数据仓库系统设计得是否合理,将直接关系到整个数据仓库系统的成败.在分析web技术与数据仓库体系结构的基础上,将XML、web挖掘技术引入到数据仓库中,构建了一种基于Web方式的分布式数据仓库体系结构.基于Web的分布式数据仓库系统的创建是一项既具有挑战性又有益的工作,与传统的数据仓库相比,具有界面友好、使用方便的优点,并且还可将企业分布在各地甚至全球的子公司、客户及企业外的数据库资源合理的引入到数据仓库中,为企业提供更有力的决策支持,大大提高企业的经济效益.本文在对数据仓库和web技术相结合研究的基础上,重点从应用角度设计和开发基于web方式的数据仓库中的关键问题.在此基础上提出了一种基于Web方式的分布式数据仓库体系结构.本文构建的基于Web的分布式数据仓库体系结构可大大减少数据传输过程中网络流量,合理实现异构数据源的数据集成,为数据仓库的开发起到一定的抛砖引玉的作用.文中详细分析了web数据的特点,以及XML、web挖掘技术,并且将数学方法应用于数据挖掘,建立了一种模型,改进了一种数据挖掘算法,提出了一种混合策略,并实现了部分算法.本文是按以下顺序组织的:第2部分简要介绍了数据仓库技术.第3部分说明了基于web方式的数据仓库系统的组成并讨论了该数据仓库特点和实现方式.本文的第4部分与第5部分讨论了数据仓库系统的设计和系统实现过程中的关键技术,这是本文的重点.最后一章是对本文的总结并提出了对基于web方式的数据仓库需要进一步讨论的问题.

2. 期刊论文 王孝成 面向Web与基于Web的数据仓库 -吉首大学学报(自然科学版)2002,23(3)

针对Web与数据仓库的结合应用,即面向Web和基于web,简要介绍了数据仓库的基本体系,分析了一种5层结构的系统,以说明面向Web应用的数据仓库的特点.根据数据仓库的发展趋势,提出了基于Web(Web-based)的数据仓库概念,并将点击流(Clickstream)数据仓库作为基于Web的数据仓库的一个特例,讨论了其体系结构.

3. 学位论文 江涛 电信企业数据仓库Web服务的设计与实现 2006

随着数据仓库技术的发展，很多电信企业都已经成功实施了数据仓库系统。电信企业的数据仓库系统已经成为企业进行决策分析的重要工具，电信企业内部的其它系统甚至电信企业外部系统也开始有访问数据仓库系统应用的需求。问题也就随之产生，由于企业内部系统异构性、紧耦合性的特点，系统间的访问非常困难，SOA的出现恰恰解决了这个问题，它使用WebServices技术有效的封装了应用实现的细节，通过一系列的标准协议开发出与平台和编程语言无关的Web服务，从而降低了应用系统的耦合性并充分利用了现有的资源。

本论文围绕数据仓库对外提供Web服务展开。首先，概要的介绍了数据仓库技术，并结合电信领域实际的数据仓库系统进行了应用分析，总结出目前需要向外界提供服务的为报表和OLAP。然后，详细研究了Web服务技术和面向服务架构，明确了Web服务的定义、实现方式以及面向服务架构与Web服务之间的关系。接着，根据对Web服务和面向服务架构的研究和数据仓库的应用分析，设计出了基于面向服务架构的数据仓库系统Web服务解决方案并根据解决方案进行了系统的实现。该解决方案主要包括Web服务包装规范的设计和Web服务注册／发现系统的设计两个部分。Web服务包装规范设计是本文的一个创新点，它包括报表和OLAP的Web服务包装规范，作者将公共仓库元模型规范中对报表和OLAP元模型的定义引入到了规范的设计中，它与描述Web服务的WSDL规范相结合，根据元模型中定义的类以及类之间的关系，定义出包装报表和OLAPWeb服务所应该定义的数据类型、消息以及操作，这种基于已有标准的设计方式使得Web服务包装规范更具规范性和通用性，包装出的服务也更容易理解。Web服务注册／发现系统的设计依据面向服务架构，该系统集成了面向服务架构中服务注册者的角色。它的用户认证功能、Web服务查找功能、Web服务注册功能以及Web服务集成功能为数据仓库Web服务提供了基础性平台。论文最后对全文作了系统的总结，并提出下一步需要进行的一些研究工作。

4. 学位论文 李广军 基于WEB数据仓库（WBDW）的设计与实现 1999

数据仓库、OLAP、数据挖掘技术是实现从机械式的简单事务处理向提供复杂分析转变的关键技术.数据仓库提供一种有利于分析的数据集成机制;OLAP提供了深度查询统计功能;数据挖掘提供智能数据处理技术;统计分析是发现数据规律的基本方法.该文主要是在数据仓库和InterNet技术的结合应用,实现基于WEB的OLAP应用两方面进行探讨.Internet技术的发展使全球范围内信息共享成为可能,人们在能够简单浏览一些静态信息之后,对于如何能够从大量数据中发现对自己有用的信息产生了浓厚的兴趣.而数据仓库技术正是为决策分析应用提供支持的一种新兴技术,但很多数据仓库是基于C/S结构的.这种结构在有利于联机分析的同时也是限制其发展的因素之一,即难以大范围使用.该文探索了B/S结构下的四层体系结构(Browser/Web Server/Application Server/DW Server)的可行性,在设计过程中我们坚持OLTP与数据仓库分离的原则,同时采用中间件(Middleware)的思想而加入应用服务器.以Client/Server模式建立数据仓库是常见的模式,我们采用基于WEB的数据仓库结构.在建立民航数据仓库的基础上,为了使OLAP应用在WEB上实现速度更快,建立订座系统多维数据库,提高动态查询速度.把SAS的统计分析应用调到后台应用服务器上运行是该文所做的主要工作,采用htmSQL,Web服务器的CGI(Common Gateway Interface),这样通过Web浏览器即可动态查询SAS数据和外部的关系型数据库;利用SAS Driver for JDBC可通过Java的最小进程Applet来查询SAS数据;通过SAS/IntrNet Application Dispatcher在Web浏览器上递交SAS时间序列分析程序到SAS应用服务器上执行,并将结果返回浏览器,这是对SAS应用在WEB上实现的重要探索.同时,我们还在WEB上实现动态报表,通过htmSQL、及Jvav Applet等方式实现动态查询.

5. 期刊论文 何震瀛. 李建中. 高宏 Web数据仓库的异步迭代查询处理方法 -软件学报2002,13(2)

数据仓库信息量的飞速膨胀对数据仓库提出了巨大挑战.如何提高Web环境下数据仓库的查询效率成为数据仓库研究领域重要的研究问题.对Web数据仓库的体系结构和查询方法进行了研究和探讨.在分析几种Web数据仓库实现方法的基础上,提出了一种Web数据仓库的层次体系结构,并在此基础上提出了Web数据仓库的异步迭代查询方法.该方法充分利用了流水线并行技术,在Web数据仓库的查询处理过程中不同层次的结点以流水线方式运行,并行完成查询的处理,提高了查询效率.理论分析表明,该方法可以有效地提高Web数据仓库的查询效率.

6. 学位论文 甘泉 基于数据仓库的Web点击流的研究 2007

Web网站每天都产生大量的数据，并且随着网络信息量的增大，在很多领域传统的数据库存储方式已经满足不了客户的需要了，那么随之而来的就是数据仓库的兴起。数据仓库是面向主题的、集成的、稳定的且随时间不断变化的数据集合，用以支持经营管理中的决策制定过程。数据仓库与数据库的不同之处在于数据库系统面向事务处理，而数据仓库系统面向分析处理。

现在点击流数据与客户信息的整合已成为WEB数据分析的最新前沿。为点击流分析而建立起的数据仓库称为点击流数据仓库。点击流数据仓库是数据仓库技术发展的一个方面。它包括了数据仓库的维度建模方法、点击流数据仓库的ETL，设计、实施和OLAP技术等方面。

与此同时计算机应用也逐渐分为了2大类：操作型处理和分析型处理，操作型处理主要是为一个组织某些方面服务的，分析型处理则用于高层管理的决策分析，也是信息处理技术的发展趋势。

在本篇论文中，作者首先对点击流数据进行提取并进行预先处理，然后确定维度和数据市集，对数据进行抽取、转换、清洗、装载，进而构建了点击流数据仓库，最后通过OLAP技术进行了分析。本论文采用的例子是湖北教育网点击流数据仓库的建模过程。

首先是点击流数据的收集和预处理。为了更好的收集点击信息，采用了在应用服务器层收集点击流数据的方法，在对用户访问会话事务的识别上采用的是最大前向参引模型和时间窗口模型相结合。

其次是教育网数据仓库要实现的基本的目标和各个维度的设计，包括确定数据市集、维度和度量值。

最后是数据仓库ETL设计，以及在数据仓库建好后，多维数据集的展示和OLAP的设计分析和实现。在ETL设计中采用数据转换服务DTS，OLAP查询分析使用了DMX的分析方式，并且具体分析了其它几种效率较高的查询分析技术。

7.期刊论文 王蔚.Wang Wei 数据仓库与Web技术的应用研究 -图书馆学研究2007(3)

本文针对数字图书馆建设过程中传统数据仓库技术应用的问题,在简单介绍数据仓库技术和Web技术的基础之上,介绍了两种技术结合的产物--Web数据仓库系统.文中主要讨论了它的体系结构、特点以及安全性等问题,并介绍了它的主要应用领域.

8.期刊论文 张明杰.Zhang Mingjie 基于Web的数据仓库安全性研究 -计算机与数字工程2010,38(7)

随着信息技术的迅速发展,特别是互联网的普及,基于WEB的数据仓库技术成为计算机领域研究的热点.首先介绍了在网络应用中数据仓库的体系结构及其优点,重点对基于Web数据仓库的安全性进行了研究,提出了相关的数据仓库的安全问题的解决办法.

9.学位论文 刘红杰 基于Web和数据仓库的决策支持系统研究 2003

随着计算机信息技术与社会经济的发展,世界经济日益呈现全球化、网络化、信息化与知识化的特征.企业每天都会产生大量重要的数据信息,然而其中仅有一小部分会在相关的业务分析中被使用,大多数企业都处于"数据过剩,信息不足"的状态.将Web和数据仓库技术、OLAP和数据挖掘技术用于现代企业管理决策中则可以大大提高企业的决策能力.其中,数据仓库侧重于数据的存储和组织,OLAP侧重于数据的分析,数据挖掘则致力于知识的自动发现.若把三者结合起来,就可以使它们的能力得到更加充分地发挥.该文首先分析了决策支持系统的研究现状及发展趋势,阐述了数据仓库系统的结构,总结了构建数据仓库的基本步骤以及数据仓库设计过程中的主要问题,并分析了数据仓库中元数据的作用.随后,又分析了Web技术的发展对决策支持系统的影响,提出了基于Web和数据仓库的决策支持系统的功能、基本框架与决策程序.最后,作者以基于Web和数据仓库的决策支持系统理论为基础,并结合证券公司的实际需求提出了基于Web和数据仓库的证券公司决策支持系统的体系结构、并进行了数据模型设计.

10.期刊论文 刘冬.Liu Dong Web数据仓库系统 -微型电脑应用2006,22(2)

本文在简单介绍数据仓库技术和Web技术的基础之上,介绍了两种技术结合的产物--Web数据仓库系统.文中主要讨论了它的体系结构、特点以及安全性等问题,并介绍了它的主要应用领域.