摘要

目前，入侵检测系统主要分为基于主机检测和基于网络检测两种，一般是对系统的审计日志进行数据挖掘或者对网络包进行截取分析，通过模式匹配等手段发现入侵。这些系统的主要问题是信息来源比较单一，容易产生漏报和误报；截取网络包消耗了较多的系统资源，影响了正常的工作。另外，管理员在利用入侵检测系统防范入侵的同时，一般还需要额外的网络管理软件对网络中的设备进行维护，这使日常管理工作变得复杂。

为了解决这些问题，在CIDF基础上提出了一种基于分布式代理的网络入侵检测系统模型（DA-NIDS）。并结合Java Applet、WWW、CORBA以及简单网络管理协议，对该入侵检测系统进行了网络管理功能的扩展。

系统在提供入侵检测功能的同时，提供浏览设备管理信息库、配置设备参数、监测设备运行参数等功能，从而简化了管理员的工作。另一方面，因为简单网络管理协议得到众多网络设备的支持，不用添加额外的程序就可以获取很多有用的信息。将这些信息提供给入侵检测系统，提高了入侵检测的准确率。运用简单网络管理协议中的消息格式进行通信，使入侵检测系统本身的安全能得到保证。对管理信息库使用基于视图的访问控制，减少了管理员的误操作。另外，借助于标准协议，系统可以方便的扩展并且能和其他符合该协议的系统协同工作。

关键词：网络安全；入侵检测；网络管理；简单网络管理协议；管理信息库；访问控制；安全模型

ABSTRACT

At present, there are two basic intrusion detection systems, host based and network based systems. They can find intrusion by means of data digging of audit records, analyzing packets from network and pattern matching. There are some problems of these systems. First, the source of information for detecting is limited, it'll course report missing or mistaking. Secondly, analyzing packets from network takes so many resources that daily work cannot be done efficiently. Above all, managers need one system for intrusion detection and need another for network management at the same time. This makes daily works complicated.

To solve these problems, a model of Distributed Agent-based Network Intrusion Detection System (DA-NIDS) is present. Combining technologies of java applet, web service, CORBA and SNMP, a new intrusion detection system with network management utilities is build.

Providing the ability of intrusion detection, the system also helps the manager with utilities for exploring the management information base, configuring and monitoring the devices. Since SNMP is widely supported by device vendors, without any extra program we can get much useful information. With this information, intrusion detection system can work more accurately. Using the message format of SNMP to communicate makes the system safety. With view-based access control of MIB, manager can work with few mistakes. Above all, with the help of standard protocol, the intrusion detection system becomes flexible for extending and universal for cooperating with other systems.

Key words: Network security; intrusion detection; network management; simple network management protocol; management information base; access control; security model

1 绪论

1.1 计算机安全、网络安全概述

随着信息时代的来临，计算机已经成为现代社会生产、生活中的不可缺少的工具。最初的计算机主要用于科学计算，它的体积庞大，运算速度也比较慢；随着科学技术的进步，计算机的体积不断缩小，运算的速度也不断提高。计算机已经从一种的专用的计算工具逐步过度到一种必备的工具。计算机应用已经逐步渗透到人类的社会活动中，并对社会的政治、经济和文化产生了深刻的影响。科学技术是第一生产力，而计算机作为科学技术的结晶，已经成为提高社会生产力的重要手段之一。并且，随着计算机网络的出现、发展和壮大，以前只有大型计算机才能完成的任务，现在只要借助于计算机网络就可以完成。计算机网络已经逐步成为党政机关、商业和金融部门、交通运输部门日常工作不可缺少的重要工具，在这些计算机网络中传递的数据不但涉及到个人隐私、商业机密等内容，甚至还有可能涉及国家政治、军事、经济的秘密，所以，计算机的安全问题就显得十分重要。“如何保证计算机中保存的数据不被窃取、篡改？”“如何保证计算机网络中传递的信息不泄露？”等等，这些问题越来越受到重视。

国际标准化组织中将计算机安全定义为：“采用一定的技术或著管理来对数据处理系统进行安全保护，包括保护计算机硬件、软件数据不因偶然的或故意的原因而遭到破坏、更改和泄露。”[1]这里是侧重于对静态信息方面的保护。而我国公安部将计算机安全定义为保护计算机的硬件、软件和数据，包括保证不受偶然或者恶意的原因遭到破坏、更改和泄露，强调整个系统能够正常运行。这里是侧重于动态意义的描述。

计算机安全涉及到的范围很广，涉及到各种影响计算机安全的因素和保障计算机系统及其正常运行的安全措施。计算机安全研究对象主要有包括：人员，即计算机系统的操作人员；计算机系统，包括计算机软件、硬件；计算机及其运行的环境，包括物理设备环境和管理环境。对于软件专业的研究人员，主要的研究对象是计算机系统本身，特别是系统软件级的安全问题。

计算机安全应在软件方面主要考虑：计算机网络系统中的通讯安全、操作系统自身的安全、数据库管理系统的安全以及应用系统的安全。

这几个方面的内容是互相补充、相互关联的。例如一个实际的应用系统中，自身有完善的安全认证、密码保护等措施，保证系统的安全，但如果它运行在一个缺乏安全措施缺乏的操作系统上，或者运用的数据库管理系统的安全级别很低，那么最终还是不能保证整个系统的安全。

计算机安全的内容应包括物理安全和逻辑安全两个方面。

物理安全主要是指系统的硬件设备和设施受到物理层面的保护。因为当前的条件下，计算机设备的费用仍然十分昂贵，所以对于计算机的网络设备、主机或者其它硬件设备，考虑它们的物理安全是一个非常必要的问题。保证物理安全的手段主要包括：保安警卫、放置设备的机房中的报警系统、钥匙或信用卡识别等。

逻辑安全主要包括信息的完整性、保密性和可用性。这里的完整性、保密性和可用性是专门针对计算机安全方面的。其中，完整性主要是指信息不会被非授权修改以及信息保持一致性等，和数据库领域的数据完整性的本质是相同的。保密性是指高级别的信息只有在授权的情况下才能流向低级别的客体与主体。可用性是指合法用户的正常请求能够及时、正确、安全地得到服务或者回应。

比较物理安全和逻辑安全两方面，可以发现物理安全比较容易实现，而且实施起来不需要很高的技术。而逻辑安全领域需要研究的范围十分广泛，这方面的研究具有巨大的潜在价值。目前对计算机安全的研究，主要的工作重点是对逻辑安全方面的研究。

总的说来，计算机安全的目标就是要建立、维护对计算机系统的控制，保证系统能够正常工作和运行。现在，可以采取多种控制方法来达到这个目标。这些方法主要分为六种：

1. 物理控制

一般来说，某些最容易、最节省和最有效的控制就是直接的物理上的控制。如前面提到的，物理控制可以包括：门禁系统、入口警卫设置、对重要软件、数据进行备份等。但是，人们在寻找更复杂的安全控制方法时，有时容易将这些简单的物理控制忽略，所以，往往简单的盗窃行为可能使系统遭受重大损失。

2. 硬件控制

现在，很多计算机安全研究机构生产了各种专用的硬件设备来保证和实现计算机的安全。这些设备包括硬件加密器、通信保密机、加密狗以及智能卡等硬件设备。比如在连接两个相距很远的子网时，先将数据通过通信保密机加密，然后送入路由器，

这样，传递的信息在传输过程中就不会被窃取。

3. 软件控制

因为计算机的使用离不开操作系统和应用程序，所以要保证这些软件的安全。计算机的操作系统本身也是由开发人员编制的程序组成，另外，操作系统之上运行的应用程序也是如此，所以，在研发这些程序代码时，要考虑到安全问题，能够排除外部攻击。软件控制主要包括：开发控制、操作系统控制、内部程序控制 和审计控制。其中，审计控制是跟踪并记录行为，利用这些记录作为事后的调查线索。

由于软件控制将影响到整个系统的可用性问题，所以必须认真的进行设计和开发工作。但由于解决安全要耗费系统的一些资源，使正常的系统工作效率降低，所以如何解决安全和系统性能之间的矛盾是值得研究的课题。现在有很多系统因为加强了软件控制，增强了系统安全的安全性，而导致系统的效率严重下降。

4. 数据加密

目前，提供给计算机中信息安全问题最有效的解决方法就是对信息加密，这种方法很早就在实际中被采用。通过数学的角度看，加密的本质就是通过对数据进行变换，使外部的观察者看不懂数据的真实意义。这样一来，即便外部的攻击者能够窃取到加密的信息，也不能获取或者更改信息的内容。

加密方法为数据提供了保密性。此外，加密还可以保证数据的完整性，因为通常读不懂的数据也难以更改。由此可见，加密是计算机安全控制中的一种重要工具，但我们也不能过高估计其作用。加密并不能解决计算机安全中的所有问题。此外，如果加密使用不当，可能降低系统的安全性或可用性。所以必须根据具体的情况考虑加密的时间、地点和采用的加密措施。

5. 策略控制

除了以上几种方法以外，有些安全控制可以通过策略控制来增强。有些策略，比如口令字的选取、定期更换等，不需要增加系统的开销也能取得很好的效果。

6. 其它控制

另外，利用法律、道德等工具来进行控制也是计算机安全中不可缺少的方法。为业务人员制定相应的道德规范并在法律上严厉打击各种计算机犯罪都是十分必要的。

一个完整的安全信息系统要综合地使用身份鉴别、访问控制、数据加密、审计、安全管理等安全技术。在实现中，没有任何一项技术能够独立的完全解决安全问题。一个设计完好的系统应该尽可能地综合运用这些技术来实现总体的安全目标。

随着信息高速公路的架设，世界电子商务的发展和我国政府机构上网工程的不断深入，越来越多的公司、企业和政府机构在网上办理业务，这种高效、便捷的方式不仅为我们提供了生活的便利，为企业创造了高额利润，同时也使得政府机构办公效率有了显著提高。但随着网络服务功能不断扩大，在提供给用户便利的同时，也给高科技犯罪提供了途径[2]。以前的单机系统一般与外界是隔离的，其中的信息不容易被获取。现在，网络黑客通过Internet，可以入侵到公司、企业甚至是政府机构的网络中，窃取、篡改计算机中的信息，或者对计算机网络进行破坏，导致网络服务瘫痪，给公司、企业和政府造成巨大的经济损失。这种情况之下，就需要对计算机网络进行有效的监视和管理[3]。

使用入侵检测系统的原因：防范网络攻击的主要常用手段就是防火墙。从技术理论上讲，防火墙已经成为一种先进、复杂的应用层网关。它不仅能完成传统的过滤任务，同时也能够针对网络应用提供相应的安全服务。利用防火墙，在正确配置的情况下，一般能够对内部子网提供有效的安全保护，防范多数的恶意攻击，从而降低网络安全的风险。

虽然防火墙的功能越来越强大，但总有一些入侵者能成功的绕过防火墙。一方面入侵攻击手段在不断发展，另一方面，来自内部子网的恶意破坏也是防不胜防。

1.2 入侵检测系统的概念

入侵检测（Intrusion Detection, ID）技术是利用入侵者留下的痕迹信息，比如系统日志、可疑进程等，发现来自外部或内部的非法入侵的技术[4,5,6]。入侵检测首先要搜集足够的信息，包括计算机系统的日志文件、计算机网络设备（如路由器）中的相关数据信息，并根据这些数据，进行一定的分析后，发现系统或网络中是否有违反安全策略的行为或恶意攻击操作[7]。入侵检测系统（Intrusion Detection System）在发现入侵行为后，可以通过预先设定的功能，发出报警通知管理员或者进行操作，将可能由入侵带来的损失减至最小[8]。

1.3 入侵检测系统的历史和现状

入侵检测技术，最早是从传统的审计技术发展而来的。在计算机发展的历史上，

早期为了对用户上机的时间进行记录并收取费用，采用了一定的审计功能。随着计算机的普及和在商业应用不断扩大和深入，审计的功能也越来越丰富，通过审计信息，用户可以了解计算机系统的使用情况，并可以利用审计信息，在系统发生问题时进行追踪、调查。计算机审计逐步分化为主要为商业公司服务的EDP审计和用于政府、军事等部门的安全审计。

二十世纪七十年代，随着计算机的运算速度不断提高、应用不断普及，对计算机安全的需求也显著增加。1980年，James P. Anderson在一份研究报告中提出建议改变计算机审计机制以便为计算机安全人员的追踪、调查提供信息，开创了入侵检测研究工作。从1984年到1986年，Dorothy Denning和Peter Neumann负责研究开发了一个实时入侵检测系统模型，系统被命名为入侵检测专家系统。在这一模型中，提出了计算机中的反常活动和不当使用之间的相关性，这成为后来入侵检测研究和系统原形的基础。1989年，美国国家计算机安全中心开发研制了Multics入侵检测和报警系统，并将系统投入实际运行。这是第一个将监控与因特网相结合的入侵检测系统。由加利福尼亚Davis分校开发的网络系统监视器在入侵检测研究历史上是一个具有重要意义的里程碑，开创了将网络流量作为主要检测数据源的先例，试图把入侵检测系统扩展到异种网络环境的系统。在此之前，多数入侵检测系统的信息主要来自于操作系统的审计日志信息以及其它以主机为中心的信息。到二十世纪八十年代末，美国空军、国家安全局和能源部资助的分布式入侵检测系统项目中，首次将基于主机的入侵检测和基于网络的入侵检测结合在一起。

目前，国外有很多实验室和公司在从事入侵检测系统的研究、开发工作，已经完成了一些原形系统和产品。如思科公司的NetRanger、Network Associates公司的CyberCop、Internet Security System公司的RealSecure等。国内的研究机构和从事网络安全产品的公司也进行了相关的研究、开发。虽然国内的入侵检测的产品较少，但有多个入侵检测产品通过了公安部、国家信息安全评测中心等机构的认证并获得销售许可证。典型的产品有东软公司的NetEye IDS、清华紫光的UNISIDS入侵检测系统等。

1.4 入侵检测系统的发展前景

网络的普及和发展的速度越来越快，互联网上的入侵、攻击手段也是层出不穷。虽然计算机信息安全、网络安全已经成为大家关注的焦点，但现在这些问题还是十分严峻。由于早期的计算机网络建设中缺乏经验，所以很多现存的网络应用中缺乏对安全问题的考虑。除了在军事、经济等要害部门或行业的应用中对安全问题有所防范外，其它的计算机网络应用很少有安全防范的能力。因为计算机系统本身就可能存在一些安全隐患或者漏洞，需要不断的升级完善，所以即便那些采用了安全防范的系统也不能获得绝对的安全。为了保证计算机应用中的信息和计算机网络的安全，除了在系统中采用防火墙、身份认证、访问控制、数据加密等方法外，还需要采用入侵检测等手段，积极防范恶意攻击[9]。

随着高科技犯罪的增多，在很多敏感部门的计算机系统中，对入侵检测的需求在不断增加[10, 11, 12]。了解系统的运行状态，查看网络的负载情况，检查各种网络设备的工作甚至是计算机应用中的异常操作等，这些对于计算机网络的管理者都有十分重要的参考价值。有了入侵检测系统作为工具，可以很容易获得这些信息，经过适当的配置后，可以有效的阻止一定的恶意攻击[13]。

虽然入侵检测系统是协助管理员对系统资源进行监视管理的有效工具[14]，很多入侵可以被自动识别和防范，但在有些情况下，还需要管理员有一定的管理知识和经验，当发现入侵活动后，采用必要、适宜的方式，维护整个系统的正常工作[15, 16]。

1.5 本课题研究的目的和意义

入侵检测系统已经成为计算机安全问题研究的重点之一。我们研究的目的在于设计并实现一套具有良好可扩充性的入侵检测系统，并将网络管理技术和入侵检测相融合，在入侵检测方面作一些尝试。

网络管理是现代计算机网络中不可缺少的部分，简单网络管理协议已经经过了三个版本的改进，变得比较成熟。很多网络设备厂商提供对该协议的支持。网络管理协议中的代理、网管站等本身就是一个分布式检测体系。虽然分布式入侵检测系统检测方法繁多[17, 18]，涉及很多领域的知识，但在体系结构上，和网络管理系统有非常相似的地方。如何利用已经存在的系统为新的应用服务是节约成本的重要环节，也是长

期以来计算机运用中面对的问题。很多公司为建成某一系统投入了大量人力、财力，随着时间的推移，新的技术很快出现，旧的设备、系统面临被淘汰的危险，而用户多数都希望添加新的功能、设备时，旧的系统能够继续服务。

基于这样的考虑，因为大多数现存的网络设备都支持网络管理协议，如果将入侵检测系统和网络管理系统集成，这将是很有意义的一项工作。并且，考虑到网络管理协议带来的通用性，不同入侵检测系统之间的协同工作也将成为可能。

1.6 论文的组织结构

下面介绍论文的组织安排：

第一章绪论部分阐述本课题的研究背景，现状、研究的目的和意义。

第二章基础知识部分阐述了防火墙技术、入侵检测技术、网络管理协议和CORBA技术的概念及特点。

第三章入侵检测系统总体设计部分阐述了入侵检测系统的设计思想，提出入侵检测系统的总体框架。

第四章网络监视与控制子系统的设计与实现部分主要阐述子系统的组成结构和各模块的功能与实现。

第五章网络监视与控制子系统的安全部分详细说明网络监视与控制子系统中面临的安全问题及解决办法。

第六章结合SNMP的入侵检测系统的设计与实现部分阐述如何利用SNMP技术实现入侵检测。

第七章总结部分主要阐述入侵检测系统以及网络管理子系统的优点，说明了本文介绍的网络管理子系统对原来入侵检测系统的改进和提高之处，并对系统今后的改进方向进行了探讨。

2 基础知识

2.1 防火墙

防火墙是现代网络中不可缺少的隔离技术，对入侵检测系统有很好的借鉴作用，在实现入侵检测系统时，还可以利用现有的防火墙为其服务[19]。

2.1.1 防火墙的概念

防火墙是建立在现代通信网络技术和信息安全技术基础之上的应用性安全技术。它是一个系统，使被保护的网络和其它网络之间相互隔离，并能够完成一些访问控制功能，阻止非法的信息访问和传递。防火墙并非单纯的软件或硬件，它的实质是软件和硬件再加上一定的安全策略的集合。入侵检测系统在一定程度上可以看做是防火墙的扩展和延伸。研究、了解防火墙的相关知识和技术对构建入侵检测系统有很好的借鉴作用[20]。

2.1.2 防火墙的技术简介

计算机网络中的防火墙是用于进行网络信息安全防范组件的总称，是网络总体安全的一部分。防火墙根据预定的安全策略控制信息出入计算机网络，阻止不符合安全策略的信息通过。防火墙本身具有较强的抗攻击能力，可以提供信息安全服务。逻辑上，防火墙是一个隔离器、过滤器、监视器。防火墙采用的技术中，包括早期的简单包过滤技术、后来出现的代理防火墙（基于电路层和基于应用层的）、比较先进的动态包过滤技术以及最新的自适应代理技术。

2.1.3 防火墙的类型

防火墙从工作的基本原理上，可以分为两大类：包过滤型和代理型。其中，包过

滤型防火墙从采用的安全策略设置上，可以细分为静态包过滤型和动态包过滤型。代理型防火墙可分为应用层网关防火墙和自适应代理防火墙。还有的防火墙采用几种工作原理结合。

1. 静态包过滤防火墙

这种类型的防火墙根据预先定义好的过滤规则，审查经过的数据报文。通过检查报头信息中的IP源地址、目的地址、采用的传输协议、端口、ICMP消息类型等，将这些与过滤规则进行匹配，依情况决定是否让其通过。

包过滤防火墙一般工作在网络层或者应用层，不需要对主机上的应用程序进行改动，可以在路由器上实现。其优点是对用户是透明的，对网络性能的影响较小，与网络通讯协议无关，易于使用。但由于包过滤防火墙在应用层下，只能利用数据包中的有限信息，无法识别出应用层的入侵。并且，由于缺少上下文关联信息，不能有效过滤无状态协议如UDP、RPC等数据包。

2. 动态包过滤防火墙

由于静态包过滤防火墙的不足，在此基础之上采用动态设置包过滤规则的方法，通过基于上下文的动态包过滤模块进行检查，提高系统的安全性能。动态包过滤防火墙在网络层不断截获流入的数据包，到一定数量后，就可以确定试图进行连接的相关状态。然后利用专用检查模块对这些包进行检查。将检查的结果保存到动态状态表中，利用这些结果对后续的数据包进行安全评估。通过检查的包可以通过防火墙。

上下文的动态包过滤防火墙提高了安全性，很容易支持未来的网络协议。但采用的直接连线使内部网络暴露在外部网络之下，危险性还是很高。由于不检查报文内容，也无法避免恶意攻击，对于UDP、RPC等协议也无法检查。

3. 应用层网关防火墙

应用层网关防火墙也被称为代理服务器，工作在应用层，应用层网关通过复制传递数据，打破了传统的客户机/服务器的两层工作模式，将客户机和服务器隔离开来。内部主机同外部主机间的通信要通过代理程序建立相应的连接映射来实现。工作原理是：当外部网络的主机B要和内部网络的主机A通信时，主机B先发一个请求给代理服务器，代理服务器检查这个用户的身份，如果是合法的用户，则进一步将请求转发给A，监控B的操作，并将A的响应转发给B；当内部网络的主机要与外部网络通信时，过程相反。

应用层网关防火墙的优点是将内部网络的网络结构隐藏起来，对外网用户是不可

见的。同时，所有的内外网间的通信都受到监控。缺点是效率不高。也不能为无状态连接提供代理。

4. 自适应代理防火墙

自适应代理技术是最近运用在商业应用防火墙中的技术。它将包过滤防火墙的高速和代理类防火墙的安全等优势结合，在安全性的前提上将速度提高。一般包括自适应代理服务器和动态包过滤器两部分，并用一个控制通道相连。

2.2 入侵检测

2.2.1 入侵的主要手段和方式

1.拒绝服务式攻击（denial of service, DOS）

拒绝服务攻击能够消耗掉一台远端主机或者网络的所有资源，进而导致合法用户的访问被拒绝。这种类型的攻击属于很难处理的安全问题，原因是这种攻击易于实现、很难预防或者跟踪[21]。具体包括的方法有：SYN flood、ping of death、teardrop、UDP flood、Smurf攻击等。

2. 占用攻击

占用攻击是通过各种可能的方式，获得目标机器的使用权限，达到占有、控制和使用目标机器的目的。具体常用的方法有：猜测用户口令、安置木马程序以及程序缓冲区溢出等。猜测用户口令，即通过使用一些字典，对系统的合法用户名、口令字反复测试，以获得合法用户的口令，从而得到系统的使用权限。特洛依木马是攻击者有意放置在目标系统中，而由系统的合法用户无意或者由系统按一定条件启动，从而在系统中为入侵者开辟使用通道并获得使用系统资源的权限。程序缓冲区溢出的攻击是攻击者以普通用户身份在目标系统中运行某些程序（有漏洞的系统程序或者恶意程序），程序缓冲区溢出后操作系统指向并执行恶意代码，从而使攻击者获得系统超级用户的权限。

3. 假消息攻击

这种攻击的主要目标是系统中配置不正确的消息，主要包括：伪造电子邮件、域名服务器高速缓存污染等。在域名服务中，本地域名服务器与其他域名服务器交换信息时并不进行身份验证，这就为入侵者提供了机会，将不正确的信息加入到域名服务

器中，把用户引向恶意提供的主机，从而造成安全问题。在邮件服务系统中，由于协议并不对邮件的发送者的身份进行鉴定，因此入侵者可以对内部客户伪造电子邮件，并声称该邮件是来自某个客户相信的人，只要在邮件中附带上可安装的木马程序，或者是一个引向恶意网站的连接，就可以实现入侵的目的。

4. 信息收集型攻击

信息收集型攻击是入侵的辅助手段，它对目标系统没有进行直接的破坏活动。这类攻击主要是搜集目标系统的基本信息（包括操作系统的类型、版本号、提供的服务以及开放的端口号等）。掌握了这些信息，可以为更进一步入侵目标系统作铺垫。具体的信息搜集攻击的手段主要有端口扫描、系统信息刺探等。

入侵的主要方式可以说都是利用现有计算机系统的弱点[22,23]（计算机软件设计中的缺陷、协议对安全问题的忽略）造成。早期计算机系统的投入很大，很多旧系统不可能被完全废弃，另外，新的系统或多或少存在这样那样的安全隐患。入侵检测系统有十分重要的作用，因而成为国内外研究和关注的重点[24,25]。

2.2.2 入侵检测的分类

从实现方式上，可将入侵检测系统分为两种：基于主机的入侵检测系统和基于网络的入侵检测系统。一个完备的入侵检测系统一定是将这两种方式结合的分布式系统。

基于主机的入侵检测系统。基于主机的入侵检测系统一般以主机操作系统提供的信息作为检测的基本信息，包括系统的日志、配置文件等[26]。一旦发现这些信息发生变化，并且变化符合攻击特征，检测系统就向管理员发出报警并做出响应[27,28]。优点：1. 非常适用于加密和交换环境；2. 接近实时的检测和响应；3. 不需要额外的硬件设备

基于网络的入侵检测系统。基于网络的入侵检测系统使用原始的网络数据包作为进行攻击分析的数据源[29]。为此，一般利用某种方式（如将网卡设置为混杂模式）来截获网络数据包。通过对数据包报头及内容的分析，确定是否与某种攻击相关。例如，端口扫描一般是有人开始攻击的预兆，通过对数据包的分析就可以检测出明显的端口扫描。有些非法攻击是通过向目标主机发送非法格式的数据包实现的，如果对报

文格式分析就能检测出这种攻击。还有些情况，通过统计数据，对网络负载情况进行比较，也可以发现一些攻击。优点：1. 成本低；2. 攻击者转移证据困难；3. 实时检测和应答。一旦发生恶意访问或攻击，基于网络的入侵检测系统能够在很短的时间内发现并作出反应，从而将入侵活动对系统的破坏降到最低；4. 能够检测未成功的攻击企图；5. 与操作系统无关。

2.2.3 入侵检测方法

入侵检测方法一般可以分为两类，即异常入侵检测和误用入侵检测。

1. 异常入侵检测

异常入侵检测又称为基于行为的检测。检测异常行为是一种较为常用的方法。在异常入侵检测中，有一个前提条件：入侵性活动集合是异常活动集合的子集。所有的入侵活动都属于异常活动。最最理想状况是异常活动集合与入侵性活动集合相等，这样只要检测到所有的异常活动，则可以判定检测到了入侵性活动。但在实际中，入侵性活动不一定体现为异常活动，而异常活动也不都是入侵性活动。两者间的关系如图2—1所示。这里存在四种可能性：（1）入侵性且异常；（2）入侵性且非异常；（3）非入侵性且异常；（4）非入侵性且非异常。基于异常的入侵检测要解决的问题就是构造异常活动集并从中发现入侵性活动子集。异常检测的方法依赖于异常模型的建立，不同模型构成不同的检测方法。异常检测是通过观测到的一组测量值偏离度来预测用户行为的变化，然后作出决策判断的检测技术。目前用于异常检测的方法主要有统计方法和神经网络方法。其中，统计法比较容易实现。

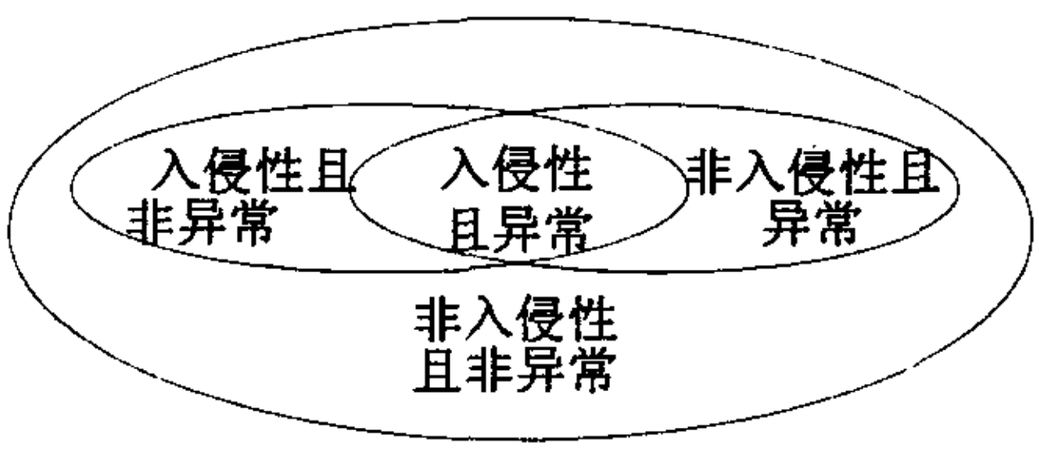

图2-1 入侵活动的几种可能情况

（1）概率统计方法

概率统计方法是基于行为的入侵检测中应用最早也是最多的一种方法。首先，探

测程序根据用户的操作为每个用户建立特定的用户特征表，然后通过比较当前的特征与原来存储的特征，判断当前进行的操作是否属于用户的异常行为。用户的特征表需要根据前面提到的系统审计记录不断的更新。用于描述用户特征的变量类型主要有：①操作频率：一定时间内操作执行的次数，通常用于检测通过长时间平均觉察不到的异常行为；②审计记录分布：在最新纪录中所有操作类型的分布；③范畴：在一定动作范畴内的特定操作的分布情况；④数值：产生数值结果的操作。

这些记录具体操作的特征可以包括：CPU 的使用情况、使用的时间和地点、内存的使用情况、打印操作、编辑器的使用情况、创建或修改文件的情况、网络上的通讯量、等。

在SRI/CSL的入侵检测专家系统(IDES)[25]中给出了一个简单的特征表结构：

<变量名，行为描述，例外情况，资源使用，时间周期，变量类型，门限值，主体，客体，值>

其中，由变量名、主体和客体确定每一个特征简表，而特征值由检测系统根据审计日志中分析的数据定期产生。这个特征值是所有有悖于用户特征的异常程度值的函数。如果假设S_1，S_2，...，S_n 分别是用于描述特征的变量M_1，M_2，...，M_n 的异常程度值，S_i值越大说明异常程度越大。则这个特征值可以用所有S_i值的加权平方和来表示：

$M=a_1s_1^2+ a_2s_2^2+...+ a_ns_n^2$，$a_i>0$，其中$a_i$表示每一特征的权重。

如果选用标准偏差作为判别准则，则

标准偏差：$\sigma=\sqrt{M/(n-1)-\mu^2}$其中均值$\mu=M/n$

如果某S值超出了$\mu\pm d\sigma$就认为出现异常。

这种方法利用了概率统计理论作为基础，在实现时比较容易。但在设计时也存在一定的问题。因为统计检测不关心事件发生的次序，如果完全依靠统计理论，很可能漏检那些和活动次序相关联的入侵事件。另外在这里需要定义报警的阈值，但设定的阈值过高，可能出现误报警的几率就会提高，阈值过低，又可能发生漏检。所以，阈值需要根据具体的实际情况设定，并要通过实际使用不断调整。

（2）神经网络方法

利用神经网络检测入侵的基本思想是用一系列信息单元（比如系统命令）训练神经单元，这样在给定一组输入后，就可能预测出输出。和前面介绍的统计理论相比，神经网络更好地表达了变量间的非线性关系，并且能自动学习和更新。由于日常工作的要求，用户的行为一般是可以预测的，不能预测的行为只占很少一部分。用于检测

的神经网络模块结构大致是这样的：当前命令和前N个命令组成神经网络的输入，其中N是神经网络预测下一个命令时所包含的过去命令集的大小。根据用户的代表性命令序列训练网络后，该网络就形成了相应用户的特征表。该网络对下一事件的预测错误率在一定程度上反映了用户行为的异常程度。

神经网络方法的能很好地处理具有随机特性的数据，不需要对这些数据作任何统计假设，有较好的抗干扰能力。但和统计中的阈值设定类似，神经网络拓扑结构以及各元素的权重很难确定，同样命令集的大小也难选取。如果命令集偏小，那么神经网络的输出不能很好的预测出用户的行为；如果命令集偏大，那么神经网络会因为涉及大量的无关数据而降低检测效率。

2. 误用入侵检测

误用入侵检测就是基于知识的检测。这种检测针对一些已知攻击方法，定义出相应的入侵模式，通过判断这些入侵模式是否出现来检测，和早期的检测计算机病毒的方法十分相似。这种方法由于依据具体特征库进行判断，所以检测准确度很高。因为检测结果有明确的参照，系统管理员可以明确的进行相应操作，同时，只要通过升级已知攻击的模式库就能防范更多的攻击。这种检测的主要缺陷在于检测和具体系统的依赖性很强，系统缺乏移植性。而且，具体入侵手段抽象成知识也有一定的困难。由于检测范围受已知知识的局限，难以检测出内部人员的入侵行为。基于知识的检测方法大致有以下几种。

（1）专家系统

专家系统是基于知识的检测中运用最多的一种方法。首先，需要将有关入侵的知识转化成if-then结构的规则。具体说，就是将构成入侵所要求的条件转化为if 部分，将检测到入侵后预定要采取的操作转化成then部分。当条件满足时，检测系统就判断出入侵行为发生，并根据定义的操作自动完成后续工作。这里，主要部分是专家系统的规则库，由描述攻击的if-then结构的规则组成，状态行为和相关的语义信息可以通过审计得到，检测器根据定义的规则库完成判断工作。在实现过程中，专家系统的检测主要面临的问题有：规则完备问题和效率问题。

我们不可能将所有的入侵方法都用if-then结构的规则描述出来，所以存在规则完备的问题，另外，如果规则数量很多，需进行判定的数据量也很大，那么分析这些数据就需要耗费很多工作。现在商业产品中一般不采用专家系统，而采用特征分析。像专家系统一样，特征分析也需要知道攻击行为的具体知识。但是，攻击方法的语义

描述不是被转化为检测规则，而是在审计纪录中能直接找到的信息形式。这样就不像专家系统一样需要处理大量数据，从而大大提高了检测效率。这种方法的缺陷也和所有基于知识的检测方法一样，即需要经常为新发现的系统漏洞更新知识库。另外，由于对不同操作系统平台的具体攻击方法可能不同，以及不同平台的审计方式也可能不同，所以特征分析检测系统进行构造和维护的工作量都较大。

（2）模型推理

模型推理是指结合攻击脚本推理出入侵行为是否出现。其中有关攻击者行为的知识被描述为：攻击者目的，攻击者达到此目的的可能行为步骤，以及对系统的特殊使用等。根据这些知识建立攻击脚本库，每一脚本都由一系列攻击行为组成。检测时先将这些攻击脚本的子集看作系统正面临的攻击。然后通过一个称为预测器的程序模块根据当前行为模式，产生下一个需要验证的攻击脚本子集，并将它传给决策器。决策器收到信息后，根据这些假设的攻击行为在审计记录中的可能出现方式，将它们翻译成与特定系统匹配的审计记录格式。然后在审计记录中寻找相应信息来确认或否认这些攻击。初始攻击脚本子集的假设应满足：易于在审计记录中识别，并且出现频率很高。随着一些脚本被确认的次数增多，另一些脚本被确认的次数减少，攻击脚本不断地得到更新。

模型推理方法的优越性有：对不确定性的推理有合理的数学理论基础，同时，决策器使攻击脚本可以与审计记录的上下文无关。另外，这种检测方法也减少了需要处理的数据量，因为它首先按脚本类型检测相应类型是否出现，然后再检测具体的事件。但是创建入侵检测模型的工作量比别的方法要大，并且在系统实现时决策器如何有效地翻译攻击脚本也是个问题。

（3）状态转换分析

这种方法的本质是将状态转换图应用于对入侵行为的分析中。在状态转换分析中，将入侵作为一系列的有序行为看待，该行为序列可以将使被检测的系统从初始的状态转入到入侵状态。首先，需要为已知的入侵方法确定系统的初始状态、入侵状态和状态转换的转换条件。然后，用状态转换图表示状态和特征事件。

我们可以利用Petri网描述入侵行为。Petri网能够形象、直观表示入侵过程，有时，复杂的入侵过程可以通过Petri网简单的表达。下面是这种方法的一个典型的例子：我们规定，在一分钟内如果登录失败的次数超过5次，那么入侵检测系统就发出警报。图2-2是该规定的Petri网表示。

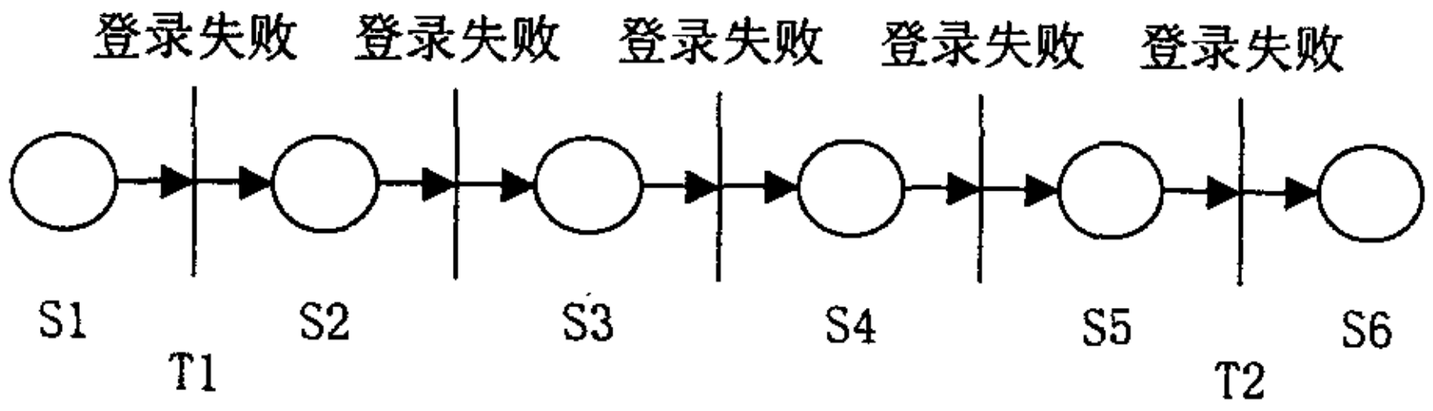

图2-2 用Petri网分析一分钟内5次登录失败

其中竖线代表状态转换，如果在状态 S1发生登录失败，则产生一个标志变量，并存储事件发生时间 T1，同时转入状态 S2。如果在状态 S5时又有登录失败，而且这时的时间 T2-T1<60 秒，则系统转入状态 S6，即为入侵状态，系统发出警报并作相应措施。

2.3 简单网络管理协议

适当的运用标准的管理协议可以使得我们的系统具有一定的通用性，只要按照协议的规定实现系统，就可能实现不同系统的协同工作。

数据网络中主要包括不同厂家的网桥、路由器、广域网链路和终端设备，用户们需要一些易于安装和操作的自动工具帮助他们对设备进行配置，并且这些工具不会给网络带来巨大负担。这种强烈的需求导致了网络管理协议的出现。

简单网络管理协议（Simple Network Management Protocol, SNMP）最早于1988年成为标准。SNMP提供对基于TCP/IP网络的管理功能，很快它成为事实上的管理标准，影响范围包括了互联网技术在内的广泛应用，很多供应商的网络设备提供对SNMP的支持。SNMP最吸引人的地方是它非常简单，因为它只提供一组关键功能，非常容易实现、安装和使用。同时，合理地运用不会对网络造成很大负担。更重要的是，简单使得内部合作成为很容易的工作：不同厂家的SNMP模块可以共同工作。

2.3.1 SNMP的工作原理和优缺点

SNMP中的三个基本组成部分是：管理站（managers）、代理(agents)和管理信息库(Management Information Base,MIB)。管理站是系统中不可缺少的，至少有一个

管理节点运行SNMP管理软件：所有需要被管理的网络设备如网桥、路由器、服务器和工作站上都要安装代理软件模块。代理主要负责提供对本地MIB对象的访问，MIB对象与被管理设备的资源和活动对应。同时，代理还负责响应管理站的命令，从MIB中取出对象的值，或者为对象设置值。通过对这些对象的操作，管理站可以了解被管理的设备运行的状况，并实现管理配置功能。例如，管理站访问被管理节点上一个追踪流入、流出数据包的整型对象，就可以了解网络中该节点的负载情况。又例如，管理站可以通过将某个表示链路连接状态的对象设置为不可用状态，从而实现使链路失效的目的。图2-3是SNMP工作的简单原理图：

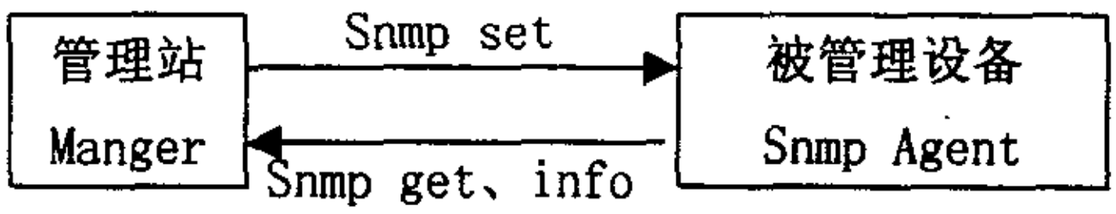

图2-3 SNMP工作原理图

这些特性对实现一个基本的网络管理系统是十分有利的。为了增强基本功能，在1993年提出了一个新的SNMP版本，在1996年进行了修订。SNMPv2增加了大数据量传送和其他一些扩展功能。但是，和很多早期的互联网协议一样，SNMP的最早两个版本都没有提供安全特性。特别是，SNMPv1和SNMPv2都没有提供对管理信息源的认证，也没有信息加密。没有认证，未经过授权的用户可能执行SNMP网络管理功能。同时，未经授权的用户可以窃听从被管理系统传送到管理站的信息。因为这些不足，很多SNMPv1/v2的实现中仅提供只读能力，减少网络监视器的能力，同时，都不支持网络控制的应用系统。

2.3.2 SNMPv3

为改进SNMPv1/v2中的安全缺陷，1998年1月提出了一系列征求意见的草案，讨论SNMPv3。这些文件没有提供一个完整的SNMP解决方案，但定义了一个SNMP全面框架和一套安全解决方案，并有意在现有的SNMPv2中应用。SNMPv3是SNMPv2加上管理和安全。

SNMPv3包括三项重要的服务：认证（authentication）、保密（privacy）和访问控制（access control）。为了将这些服务用一种灵活、有效的方式展现，SNMPv3提出了演员（principal）的概念。实体将以演员的名义，获得服务或者处理一些任

务。演员可以是扮演特定角色的个体，可以是一组个体（其中每个都扮演特定的角色），或者是一个应用或一组应用，也可以是这些的组合。本质上看，一个演员在管理站上操作，向代理系统提出SNMP命令。演员的身份和目标代理一起决定涉及到的安全特性，包括认证、保密和访问控制。演员概念的引入使安全策略适用于特定的演员、代理和信息交换，并使得安全管理员在为用户分配权限时有一定的灵活度。

SNMPv3是用一种模块化方式定义的。每个SNMP实体都具备唯一的SNMP引擎（engine）。SNMP引擎中实现了接收/发送消息、认证和加/解密消息以及对被管理对象的访问控制功能。这些功能通过服务的形式提供给一个或者多个应用程序。这些应用程序因为都配置了SNMP引擎而成为SNMP实体。这种模块化方式有很多优点。首先，SNMP实体的角色由该实体中实现的模块决定。例如，一个SNMP代理中需要一些特定的模块，而一个SNMP管理站可能需要另外一些特定模块。其次，规范中的模块化结构使得规范本身可以定义每个模块的不同版本。这也就是说，当定义一个替换或者增强某些SNMP功能时不需要全新的SNMP版本（如SNMPv4），而只需要描述共性和策略上的改变。SNMPv3的模块结构如图2-4所示。

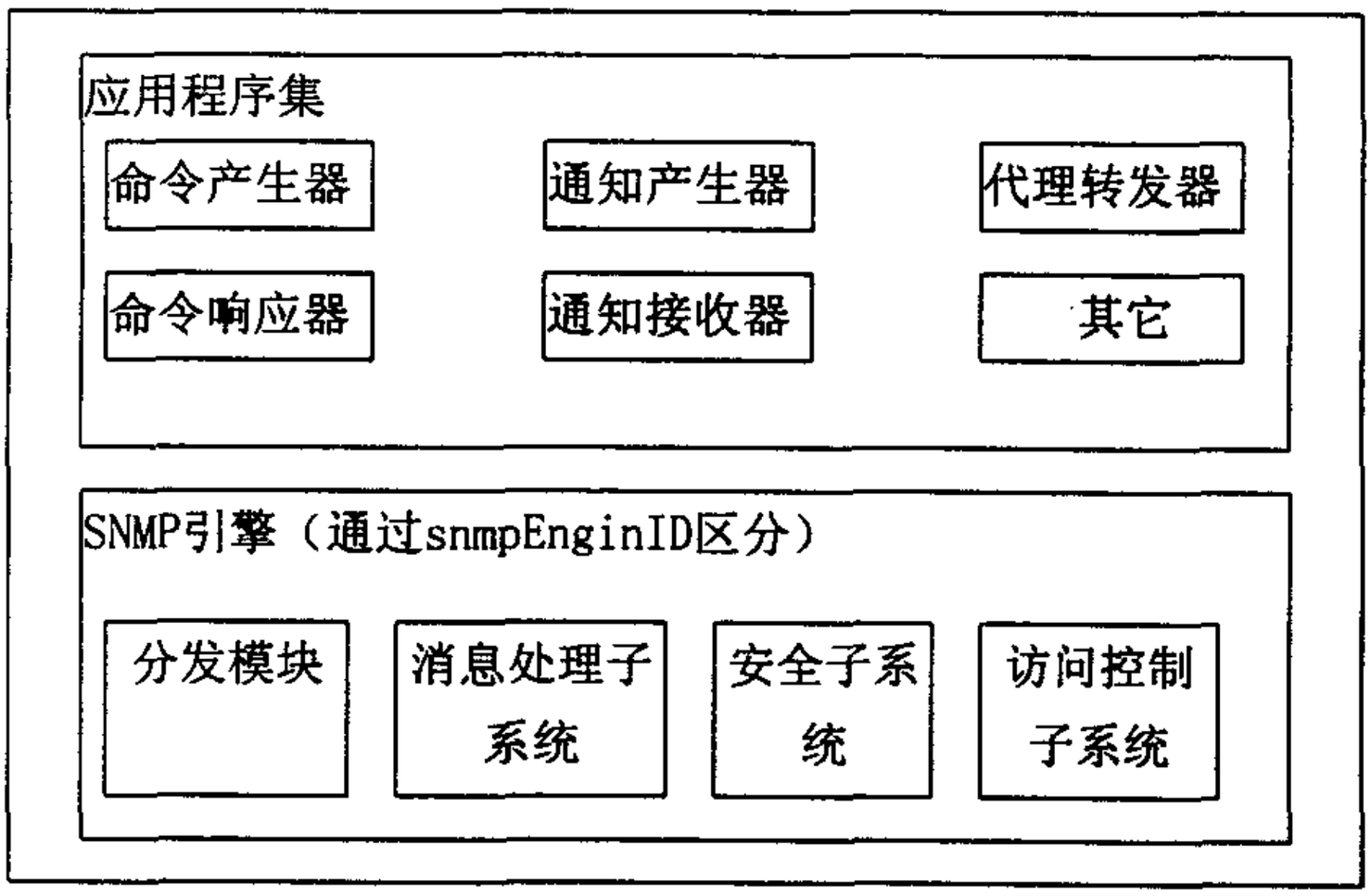

图2-4 SNMPv3的模块化结构图

各部分的说明如表2-1所示。

表2-1 SNMPv3各个模块的说明

分发模块 Dispatcher	允许在SNMP引擎中同时支持不同版本的SNMP消息。负责接收应用程序要通过网络发送的协议数据单元（PDU）并将流入的PDU分发给应用程序；将要向外发出的PDU交给消息处理子系统形成消息并将外来的消息交给消息子系统以获取PDU；在网络中接收和发送SNMP消息。
消息处理子系统 Message Processing Subsystem	负责准备待发送的消息或者将收到消息中的数据取出。
安全子系统 Security Subsystem	提供安全方面的服务如认证和加密消息。这个子系统可以包括多个安全模型。
访问控制子系统 Access Control Subsystem	提供一系列授权认证服务，使应用程序用来检查访问权限。当进行读取或修改命令时，或者产生通知时，就需要涉及到访问控制。
命令产生器 Command Generator	发出SNMP Get、GetNext、GetBulk、Set命令的PDU，并处理这些它发出命令的响应。
命令响应器 Command Responder	接收SNMP Get、GetNext、GetBulk、Set命令的PDU。查看PDU中的参数contextEngineID是否和接收该PDU的本地系统的引擎号一致，决定是否接收。命令响应器中要执行恰当的协议操作（利用访问控制），产生响应消息发送给命令源。
通知产生器 Notification Originator	监视某个系统的特定事件或者条件，基于这些事件或条件产生Trap或Inform消息。通知产生器中要有决定这些消息发送给谁的机制，并决定发送消息时用什么SNMP版本，加什么安全参数。
通知接收器 Notification Receiver	监听通知。当获得Inform消息时产生响应消息。
代理转发器 Proxy Forwarder	转发SNMP消息。代理转发器的实现是可选部分。

2.3.3 SNMPv3的消息处理

SNMP中消息处理机制有一定的安全措施，对实现我们系统的安全特性有很大帮助。

SNMPv3利用基于用户数据包协议（User Datagram Protocol，UDP）或者其他传输层协议来传递SNMP信息。在UDP层，SNMP功能由两个应用层组成：PDU处理层和消息处理层。PDU处理层是最上层。这一层，管理命令（如Get、Set、Trap、Inform）PDU实现，包括命令类型标识和一组命令指向的参数（管理对象）。这一PDU被交给消息处理层，加上消息头。消息格式的头中包含了安全相关的信息，用于认证和加密操作。

2.4 CORBA的基本概念和原理

我们的基本系统采用的是分布式结构。同时，在我们的网络监视与管理子系统中，采用了简单的CORBA机制。这种机制保证了远程监视和管理功能的实现。

公共对象请求代理体系结构（ Common Object Request Broker Architecture，CORBA）允许分布式应用程序之间进行互操作（即应用程序与应用程序之间通信），不管这些应用程序采用什么语言编写或驻留在什么地方。

CORBA规范被对象管理组（ Object Management Group，OMG）所采纳，致力于解决开发分布式对象应用程序的复杂性与高成本问题。CORBA采用面向对象方法创建在应用程序之间可重用与可共享的软件组件，每一个对象封装了它内部工作的细节，并提供一个定义良好的界面，从而降低应用程序的复杂性。由于一旦实现并测试一个对象后，它可以反复地被使用，因而减少了开发应用程序的成本。

CORBA体系结构的主要目的是定义一个描述客户机如何能向远程的对象实现发送请求的框架，并潜在地从对象处得到回应。对象接口用与编程语言无关的接口定义语言描述。基本上有两种不同的方式来让客户机和对象实现发送和接收请求：静态方法和动态方法。静态方法要求所有的IDL接口在编译时已知，这样IDL编译器就能生成桩和框架代码，这些都必须链接到实现。动态方法使用户在编译时不用了解不同的IDL接口就能实现用程序来处理任何类型的请求。处理请求的动态方式要求在客户端使用动态激发接口（ Dynamic Invocation Interface，DII）以及在服务器端使用动态框

架接口（ Dynamic Skeleton Interface, DSI）。DII和DSI通常用来构建如桥接器等一般的系统级组件。

对于正常的应用程序，静态桩和框架的使用更加普遍。静态方法的好处是使用户能很好地使用CORBA对象，就好像它们是编程语言中的普通元素一样。在客户端，这通过使用代理对象来完成。代理是远程目标对象的本地代表。代理包含足够的信息来向远程目标对象发送请求，封装网络地址、端口号等细节。代理对象通过使用客户机编程语言的标准类型，提供了以类型安全的方式来访问目标对象的方法。如果客户机想使用实现了Stock IDL接口的对象，桩代码就会向它提供等价的用特定编程语言编写的Stock接口，例如C++的Stock类。如果客户机想向远程Stock对象实现发送消息，它只需简单地激发本地代理的一个方法。桩代码，即生成的代理实现，负责打包（marshal）请求的参数，这样客户机的ORB运行时模块就能向目标服务器发送消息。服务器的ORB运行时模块读取从网络传来的消息，并把消息传给生成的框架代码，使得框架代码解包请求的参数，这样它就能把这些参数传递到目标对象的实现。生成的框架把请求作为服务器端的正常方法调用来传递，使得客户机和服务器都像对待普通的编程语言对象一样来对待CORBA对象。请求的回答能以同样的方式发送回客户机。

因为静态接口比动态接口使用得更普遍，这里着重讲述静态接口。图2-5总结了使用静态接口的CORBA远程激发的原理。图2-5中的对象请求代理（Object Request Broker，ORB）将一个客户应用程序连接到它想使用的对象。客户程序不必知道与它通信的对象实现是驻留在同一计算机中，还是位于网络上的某一台远程计算机中。客户程序只须知道对象的名字并了解如何使用对象的界面，由ORB负责查找对象、请求路由选择以及返回结果等细节。

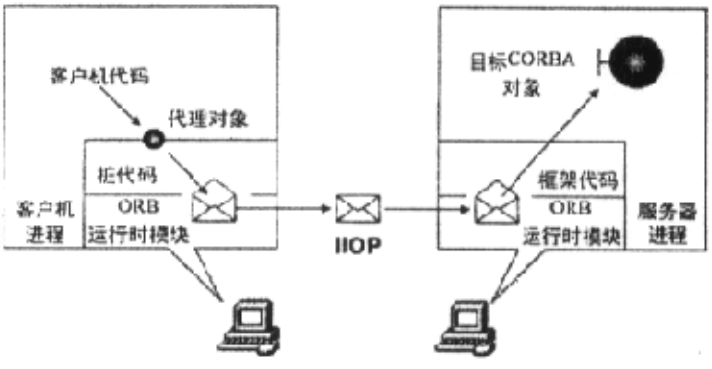

图2-5 使用静态接口的CORBA远程激发的原理

图2-6为CORBA的请求模型。请求的CORBA模型假定每个请求都有一个目标、一个操作和一系列参数。目标标识目标对象，操作描述所激发操作的名称，而参数则是需要传递的数据。一个请求必须提供一种激发功能。激发功能可以有不同的语义，例如阻塞和非阻塞的调用，或是单向的语义。

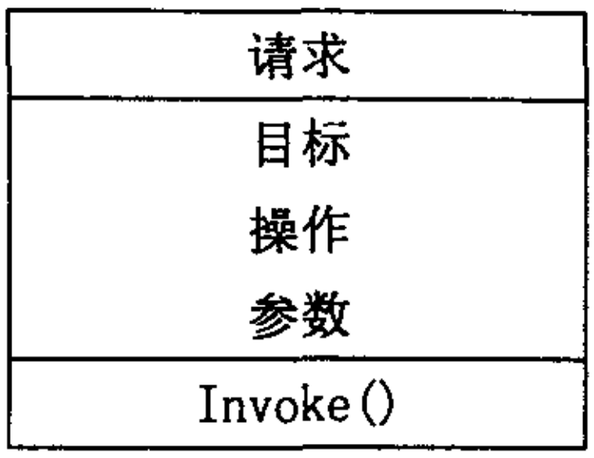

图2-6 CORBA请求模型

2.5 小结

防火墙技术是比较成熟的技术，有很多思想可以作为入侵检测系统的借鉴；分布式系统如网络管理系统、域名服务系统等与我们要研究的入侵检测系统有很多相似点，可以作为设计的参考；利用CORBA技术很容易实现分布式控制管理系统。

3 入侵检测系统的总体设计

3.1 基本入侵检测系统的设计思想

从实现方式上，入侵检测系统一般分为两种：基于主机的入侵检测系统和基于网络的入侵检测系统。两种系统针对的攻击的侧重点有所不同，有各自的优势和缺点[30]。我们提出的基于分布式代理的网络入侵检测系统属于后一类型。基于代理的检测模型有很多[31]，是一种比较通用的分布式系统的构造方法。我们在分布式环境中的多台主机上安装了入侵检测探测器，用来搜集信息，通过代理间信息的传递、汇集和分析，进一步发现高层次的网络入侵。

在分布式入侵检测系统的设计中，需要考虑很多问题，比如协作检测、系统内部安全等[32,33]。除了运用一些检测方法实现基本的入侵检测系统外，我主要考虑的问题是：如何在入侵检测系统之上，集成网络管理功能，并能同时支持不同厂家的网络设备；如何采集网络设备中的信息，运用到入侵检测系统中；如何保证网络管理部分的安全。鉴于以前无线网络管理的一些经验，我决定采用简单网络管理协议，搜集网络设备中相关的信息，在进行分析处理后，通过通信代理将结果交给入侵检测系统；建立与入侵检测系统相关的管理信息库，利用简单网络管理协议，对信息库的变量进行操作，从而实现对入侵检测系统的配置；通过动态载入不同设备的管理信息库，实现对不同设备的支持；利用SNMPv3协议的安全特性保证网络管理的安全。

3.2 入侵检测系统的组成

参考CIDF规范以及一些现有的检测模型[34,35]，我们首先提出了一种基于分布式代理的网络入侵检测系统（DA-NIDS），该系统主要分成如下几个模块：探测代理、分析代理、通讯代理、回应代理、存储代理、控制端中心。各模块的主要功能说明如下：

探测代理：探测代理的主要作用是对某一网段的数据包进行过滤，将网络数据包中可疑的数据以一定的格式的数据传送给分析代理。

分析代理：分析代理是入侵检测的关键部分，它的主要作用是对探测代理发送而来的网络数据包数据进行分析，计算出相应的警报等级，将分析后的数据和原始的数据包数据重组成规定的攻击事件格式，传送给通讯代理使用。分析代理注重于高层次的分析方法，如前面介绍过的基于统计的分析方法、基于神经网络的分析方法等。同时负责对分布式攻击进行检测。分析代理是整个入侵检测系统的核心部分。因为各种分析方法都有针对性，所以，检测时系统要结合多种方法，达到能够通过配置后可以动态更换的目的，从而减少系统的误报和漏报。

通讯代理：通讯代理的主要作用是将分析代理得出的攻击事件数据进行一定的加密保护，并通过控制端中心进行身份验证。另外，负责将事件数据传送到控制端中心。

控制端中心：控制端中心的主要作用是提供给入侵检测系统的管理员一个控制界面，对代理进行控制和管理，另外，提供对各攻击事件的查询和分析的能力，并能根据需要展示给定时间段内的相关数据的报表。

回应代理：回应代理提供对攻击事件作出相应回应的能力，响应包括消极的措施，如给管理员发电子邮件或消息等。也可以采取保护性措施，比如切断入侵者的连接请求、修改路由器的访问控制策略、修改防火墙的规则等。

存储代理：存储代理提供对攻击事件的数据的存储。同时也提供给控制端中心进行查询。它存贮探测代理捕获的原始数据、和分析代理分析的重要数据。储存的原始数据在对发现入侵者进行法律制裁时提供确凿的证据。存储代理也是不同部件之间数据处理的共享数据库，为系统不同部件提供各自感兴趣的数据。因此，存储代理应该提供灵活的数据维护、处理和查询服务，同时也是一个安全的日志系统。

3.3 网络管理协议与入侵检测系统结合

在基本入侵检测系统的基础之上，我将SNMP代理设置在各个功能模块的主机上，添加相应的调用接口，使SNMP代理能够接受标准的网络管理消息，并完成对各个功能模块的监视或者控制功能。在控制服务中心上添加WWW服务，利用Java Applet结合浏览器完成远程控制功能。由于利用了标准的SNMP协议，可以搜集不同厂家的网络设备中我们关心的参数信息，同样将这些信息交给合适的系统部件进行分析，可以帮助进行入侵检测分析。在基本入侵检测系统的基础上，将SNMP协议与之结合，形成新的具有网络监视和管理功能的入侵检测系统。图3-1是系统扩展后的结构图。

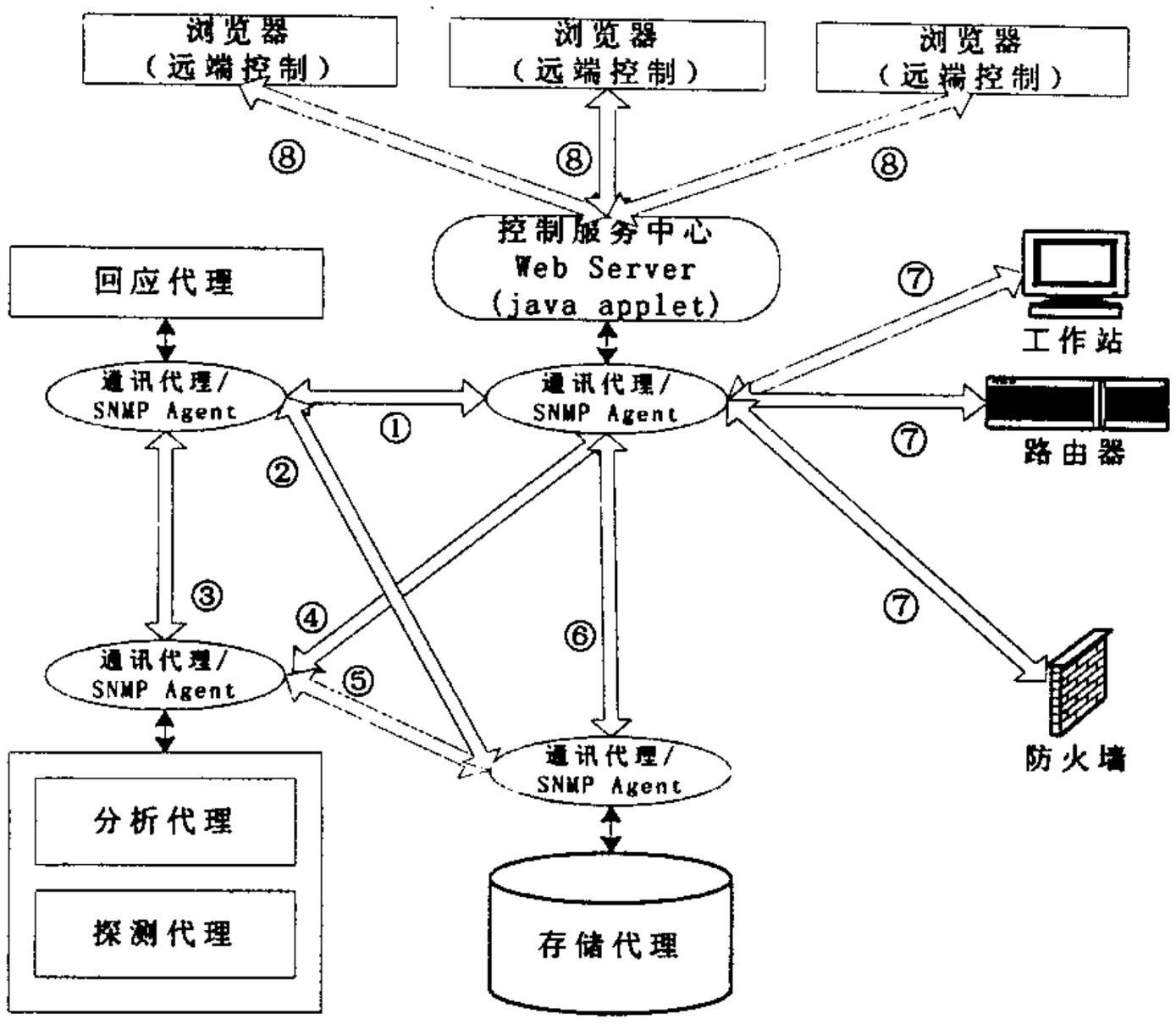

图3-1 具有网络监视与控制功能的入侵检测系统结构图

图3-1中，控制端中心向回应代理发送控制命令，而同时，回应代理向控制端中心发送响应信息，二者之间的数据流用①表示；②表示：代理从存储代理中提取回应配置信息，将响应日志信息发送至存储代理保存的数据流；分析代理和探测代理设计为集成在一起，它们向回应代理发送事件特征数据，交由回应代理采取适当的回应措施，二者之间的数据流用③表示；分析代理和探测代理接收来自控制端中心的控制命令，并同时发送代理状态信息到控制端中心以提供给管理员，二者之间的数据流用④表示；分析代理和探测代理将得到的检测信息发送给存储代理加以存储，并同时从存储代理中得到相关的配置数据，二者之间的数据流用⑤表示；控制端中心从存储代理中获得系统的配置数据和事件信息供管理员察看，同时控制端中心将控制信息发送到存储代理，二者之间的数据流用⑥表示；网络监视和管理器 通过简单网络管理协议控制网络设备，二者之间的数据流用⑦表示；管理员通过浏览器远程使用网络管理中心的WWW服务，二者之间的数据流用⑧表示。

需要说明的是网络监视和管理器为一台Linux服务器，提供WEB服务，它和原来的控制中心可以安装在不同的主机上，也可以集中在一台主机上。控制服务中心支持java applet、CORBA和SNMP。被管理的不同设备都必须支持SNMP，安装有SNMP代理。

3.4 任务分工

基本的入侵检测系统主要由同组的何中华、卢刚和刘刚同学完成，入侵检测系统的网络管理功能的扩展由我负责。我的工作主要包括：

1. 安装网络监视和管理服务器。在Linux平台上，安装WWW服务器、CORBA服务器、MSQL数据库，并提供对java的支持；
2. 为服务器增加简单网络管理协议中管理站的功能；
3. 为通信代理增加简单网络管理协议中SNMP代理的功能；
4. 建立网络管理数据库；
5. 通过Java applet实现对不同设备的远程监视和管理；
6. 利用网络管理协议的支持，采集设备信息，完成相关的入侵检测功能；
7. 保证网络管理系统的安全。

3.5 小结

将网络管理与入侵检测集成在一起，对入侵检测系统本身是重要的补充。由于利用了简单网络管理协议，可以从更加广泛的渠道搜集网络相关的信息，从而提高入侵检测的可靠性。同时，使用标准的协议，为以后的系统扩展、推广以及不同系统间的合作建立了良好的基础。同时支持多种设备、远程管理等功能使管理员的工作更加方便，但也会带来一定的安全隐患。如何解决好系统中的安全管理问题是本文的重点之一。SNMPv3中，在安全性方面有比较完善的解决方案，它使整个系统的安全问题能得到一定程度的解决。

4 网络监视与管理子系统设计与实现

4.1 网络监视与管理子系统的结构

实现网络管理系统都有一定的要求和规范[36, 37, 38]，很多硬件开发商都提供各自的网络管理系统。借鉴这些现有的产品，设计出网络监视与管理子系统的结构，其简单表示可以参考图4-1。

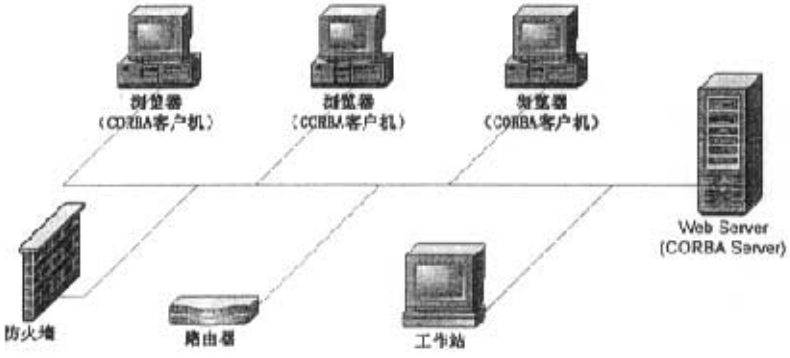

图4-1 网络监视与管理子系统的结构示意图

为了能够实现管理员的远程监视管理，我们采用了Java applet 与简单CORBA 机制结合的方式实现该子系统。图中，WWW服务器实际上是网络监控管理中心服务器。远程管理员通过浏览器下载并运行Java applet 小程序，与管理中心交互，通过CORBA机制，调用服务器上的与网络管理和监控相关的对象，这些对象通过SNMP的消息机制，从被管理的那些设备（如防火墙、路由器、网桥、工作站等）获取信息并进行适当的控制。被管理的设备必须支持SNMP协议。同时，这些搜集到的信息将通过网络监控管理中心，交给入侵检测系统使用（如写入共享的数据库等）。在入侵检测探测器中集成SNMP的管理时，用户可以通过远端的任意主机对整个入侵检测系统进行监控。这样做的同时，也将带来一定的安全问题。比如如何验证用户的身份等。

4.2 网络监视与管理子系统的基本功能模块

子系统的基本功能模块结构如图4-2：

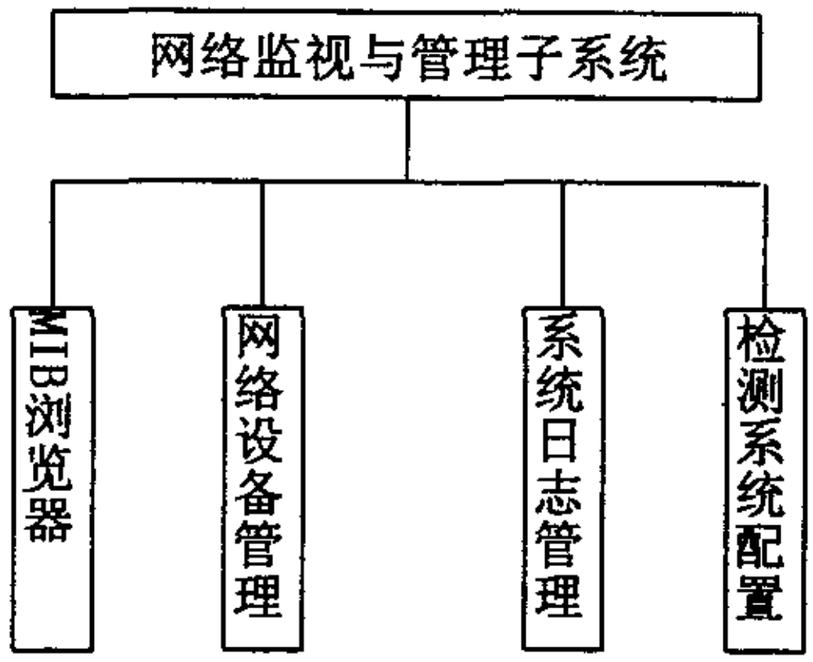

图4-2 网络监视与管理子系统基本功能模块结构图

网络监视管理系统主要由MIB浏览器、网络设备管理器、系统日志管理、检测系统配置几个部分组成。

MIB浏览器主要功能：按照协议，将符合规定格式的管理信息库以有层次的树结构展示给管理员。管理员可以通过浏览MIB子树，了解各变量的含义和功能。同时，在一定的条件下，可以用SNMP协议访问被管理设备相应的管理信息变量的值，从而达到监控被管理设备的目的。

网络设备管理器主要负责将被管理的设备常用的一些信息记录到历史数据库中，作为参考值。管理员可以定期取出一些设备最新的值，参照历史记录进行比较。通过这种途径，可以了解系统运行的轨迹。

系统日志管理主要是从远程控制端浏览WWW服务中心记录的与系统相关的信息。这些信息可能是入侵检测的结果或者被管理设备的一些性能状态等。了解这些信息，可以知道系统的运行情况。也可以将报警信息存放在日志中。

检测系统配置功能是为系统管理员提供的管理工具。通过该接口，管理员可以对整个系统的资源进行管理[39]。同时，可以向安装SNMP代理的入侵检测部件进行一定的管理。

4.3 MIB浏览器

MIB浏览器的主要功能是：将标准的管理信息库文件通过Java程序预处理，以树形结构的形式通过浏览器展示到管理员面前。用户可以选取符合规范的任何管理信息库文件，通过URL方式导入。能否执行相关操作要由用户扮演的角色、SNMP协议规定和被管理设备的支持共同决定。MIB浏览器的结构如图4-3

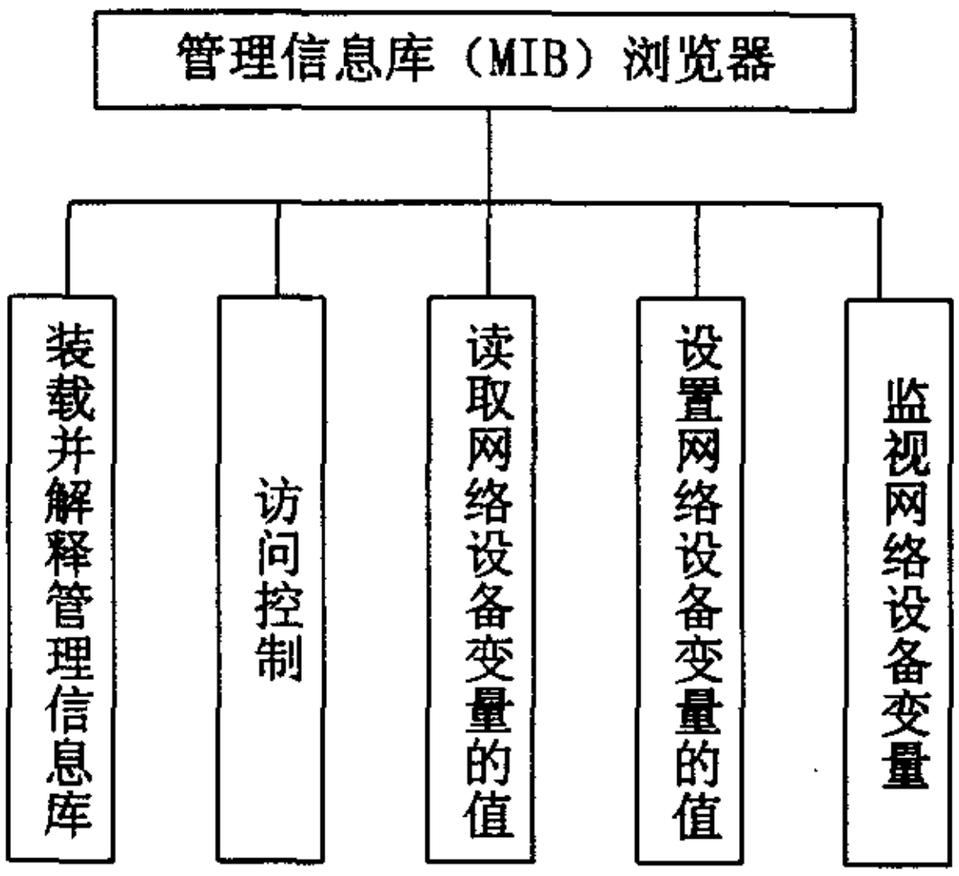

图4-3 MIB浏览器结构图

其中，装载解释管理信息库的主要步骤为：

1. 用户通过浏览器加载JAVA APPLET，输入需要加载的管理信息库的URL地址（一般由网络管理服务器以文件的形式提供）；

2. 由管理服务器读取文件，传给APPLET。APPLET依据SNMP协议标准，分析解释管理信息库，保存到JAVA类中，传到浏览器；

3. 根据预先的设定，由访问控制决定用户能够看到的管理信息库的部分。

读取网络设备变量的值的主要步骤为：

1. 用户选择要管理的设备的IP地址；

2. 从所能看到的管理信息库中选择需要查看的对象；

3. 系统根据用户选择的对象名，填写相应的对象标识串（OID）；

4. 调用远程的CORBA对象的方法，读取对象的值，并将结果传到浏览器。其中，能否读取到对象的值由被管设备中的SNMP代理中的访问控制决定。

设置网络设备变量的值的步骤和读取过程基本相同，用户要填写变量的新值。需要说明的是，MIB的树形视图中提供了对象的说明信息、类型和一般的访问策略，用户可以参考这些信息，对设备进行管理。另外，还提供给用户对某一特定的对象进行监测的功能，其实质就是对同一个对象进行反复的读取操作，利用时间-值的曲线描绘变化过程，一般只对动态对象（如IP报文流入、流出的数量）进行这种监测。

访问控制的方法与被管设备中采用的基于视图的访问控制类似。见本文的5.4节。另外，这里的访问控制需要查询用户-设备对应表，决定用户是否能够管理某一设备。

4.4 网络设备管理

网络设备的管理主要思想是：将被管理的设备按厂家和类型归类管理，在对设备进行初始配置并投入使用后，通过网络设备管理工具，记录一些设备的特征值，通过定期检查这些特征值，可以发现设备中存在的问题。网络设备管理的主要步骤为：

1. 浏览某一类设备的管理信息库，参考对象的说明，搜集该网络设备的特定参数（如系统运行时间、设备的位置、联系人、路由表、噪音等）；
2. 为该类设备建立特征值数据库；
3. 读取设备特征的初始值，存入数据库；
4. 定期读取设备的特征值，进行比对。

例如，某设备的路由表被改变，而管理员没有进行过这项操作，那么就可能有非法的入侵者进行了破坏；又比如，系统的运行时间的实际值为3小时20分10秒，而原来的记录为10小时8分20秒，则说明如果在3小时20分10秒前，没有发生停电或者人为关闭设备，那么设备可能发生过故障，或者有人恶意破坏。

这种方法，在入侵检测中可以归入完整性检查。管理员通过了解这些有一定规律的设备参数，可以掌握设备的运行情况，及时发现问题，使系统保持正常工作。

4.5 系统日志管理

系统日志管理的主要工作是将网络监视、管理系统中需要报告给管理员的信息记录下来，还包括将监测设备中参数异常值记录下来。并提供功能，将这些信息按主机、时间段等条件导出到网络管理服务器，形成日志文件。另外，管理员通过查询日志，

可以了解到以前设备的运行情况。

4.6 检测系统配置

检测系统配置是通过网络管理服务中心向入侵检测系统中被管理模块发出SNMP消息，由模块中的SNMP代理解释消息并执行相关变量的维护工作。我主要针对模块中的探测代理部分进行探测规则的配置。

首先，根据探测代理的需要，设计MIB对象。例如：需要配置代理的入侵检测管理控制中心的地址和端口，相应的MIB设计为如下，其中以“//”开头的为注释部分。

```
HUST DEFINITIONS : : = BEGIN
//定义学校HUST的MIB，BEGING为MIB的开始，最后以END结束
        IMPORTS
        enterprises             FROM RFC1155-SMI
        OBJECT-TYPE             FROM RFC-1212;
      //该MIB库文件的插入位置和参考标准
DisplayString : : =   OCTET STRING
MacAddress : : = OCTET STRING (SIZE (6))
MacUid     : : = OCTET STRING (SIZE (8))
Uchar      : : = OCTET STRING (SIZE (1))
GenAddress : : = OCTET STRING (SIZE(12))
//自定义的数据类型
HUST       OBJECT IDENTIFIER : : = { enterprises 1800 }
//HUST的OID是在插入点后值为1800
COMPUTERDEPARTMENT          OBJECT IDENTIFIER : : = { HUST 4   }
//计算机系在HUST后值为4
CS510   OBJECT IDENTIFIER : : = { COMPUTERDEPARTMENT 6 }
//实验室在计算机系后，值为6
AIDS       OBJECT IDENTIFIER : : =     { CS510 1 }
//入侵检测系统在实验室后，值为1
server OBJECT-TYPE
```

```
        SYNTAX  STRING
        ACCESS  read-write
        STATUS  mandatory
        DESCRIPTION
        "入侵检测中心控制器的IP地址"
    : : = { AIDS 1 }
```

//入侵检测中心控制器的IP地址，变量名为server，类型为字符串，可读/写，在AIDS后，值为1

```
    port OBJECT-TYPE
            SYNTAX  INTEGER
            ACCESS  read-write
            STATUS  mandatory
            DESCRIPTION
            "入侵检测中心控制器的端口"
    : : = { AIDS 2 }
```

//控制中心的端口号，变量名为port，类型为整数，可读/写，在AIDS后，值为2

```
    END
```

//MIB文件结束

比如控制中心服务器地址为192.168.1.6，端口为888，则server=192.168.1.6，port=888。通过SNMP read/write命令，可以实现对代理的配置。

4.7 小结

网络监视与管理子系统提供给用户的功能十分便捷。通过浏览器，用户不用安装任何其他程序，就能够监视网络资源的使用情况，并实现简单的管理配置功能。

5 网络监视与管理子系统的安全

整个系统可能有多个部分涉及到安全问题[40]，如数据库存储与访问[41]、用户身份验证[42, 43, 44]、代理间通信和系统整个的访问控制等。本文中的网络监视与管理子系统的安全主要包括浏览器到网络管理服务器和网络管理服务器到被管理设备两部分的安全。前一部分的通信安全部分可以用SSL协议或者一般的通信加密机制实现[45]。本文中主要讨论如何控制管理员对不同设备的访问、对同一设备不同参数的访问以及网络管理服务器到被管理设备的安全。我们将运用基于视图的访问控制来实现对管理员访问不同设备及同一设备不同参数的控制，运用基于用户的安全模型实现从网络管理服务器到被管理设备的安全。在接下来的5.1节到5.3节中，主要讨论基于用户的安全模型要解决的问题、主要目标和安全模型的结构，在5.4节中，讨论基于视图的访问控制以及如何控制管理员的访问，在5.5节中，简要论述用CORBA机制如何解决从浏览器到网络管理服务器的安全问题。

5.1 基于用户的安全模型面临的威胁

1. 安全模型主要对抗的威胁

（1）信息篡改

一些未经授权的实体可能更改网络中流过的SNMP消息，产生授权人的消息，从而实现对被管设备的未授权管理操作，包括伪造管理信息库中某一对象的值等。

（2）伪装

一些用户原本缺乏某些授权的操作，他们伪装成另外的拥有这些经过授权的操作的用户，实现非法操作。

2. 次要的威胁（模型提供有限保护的）

（1）偷听、泄露

偷听被管代理和管理站间的信息交换。对此的保护作为本地策略考虑的范围。

（2）消息流被篡改

SNMP协议是典型的基于无连接传输服务的。消息的重组、延时、重发在操作中自

然出现。消息流被篡改是指消息被故意地重组、延时、重发，以影响未经授权的操作。

3. 安全子系统不需要考虑的威胁（模型不提供保护的）

（1）拒绝服务攻击(Denial of Service, DOS)

（2）报文分析（Traffic Analysis）

5.2 基于用户的安全模型的主要目标

安全模型主要目标有：

1. 完整性保护：提供验证每个收到的SNMP消息在网络传输过程中未被篡改的能力。

2. 防止假冒：提供验证每个收到的SNMP消息是否为某个用户产生的能力。

3. 防止重放：提供验证每个收到的SNMP消息（请求或者包含管理信息的消息）是否不是当时产生的能力。

4. 信息加密：提供保护每个收到的SNMP消息的内容不被泄露的能力。

安全模型中为了达到上面的目标，将分三个模块分别完成相应的工作：

1. 认证模块：数据完整性、数据源认证。

2. 时间模块：防止远远超出正常操作时间的延时或重播。

3. 加密模块：防止消息内容的泄露。

5.3 基于用户的安全模型的结构（USM）

现在有很多访问控制模型，比如基于角色的访问控制（RBAC）[46,47]。由于应用环境比较简单，我们采用的是基于用户的安全模型（User-Based Security Model，USM）。图5-1说明了网络管理系统如何利用安全模型实现系统的安全。

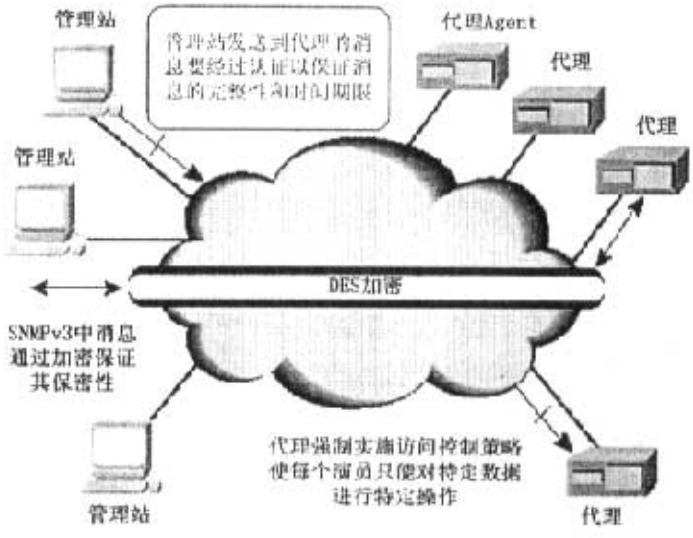

图5-1 具有安全策略的网络管理系统的示意图

在基于用户的安全模型中利用了认证引擎机制。在任何消息处理中，根据下面的规则决定发送或接收的一方是经过认证授权的SNMP引擎：

当一个SNMP消息包括的内容需要一个回应时（例如：Get、GetNext、GetBulk、Set、Inform的PDU），那么该消息的接收者就是经过认证的。

当一个SNMP消息包括的内容不需要回应时（例如：SNMPv2-Trap、Response、Report的PDU），那么该消息的发送者就是经过认证的。

因此，命令产生器发送的消息和通知产生器发出的通知消息（Inform），消息的接收者被看作是经过认证的；命令响应器发出的消息或者通知产生器发出的陷阱消息（Trap），消息的发送者被看作是经过认证的。这种指定有两个目的：

1.消息的时间期限由认证引擎的时钟决定。当一个认证引擎发出一个消息（Trap、

Response、Report）时，它包含了引擎的当前时钟，未认证的接收者就可以与该时钟同步。当一个未认证的引擎发出消息（Get、GetNext、GetBulk、Set、Inform）时，它包含了目的引擎时钟的预测值，使得目的引擎可以检查消息的时间期限。

2.用一个密钥本地化过程，可以使一个演员的密钥存放在多个引擎中[48]；这些密钥被认证引擎进行了本地化，演员与唯一的密钥对应，但分布式网络中也不用保存同一个密钥的多个拷贝。

图5-2是SNMPv3中的消息结构图。其中最先的五个字段由消息处理模型加在流出或者流入的消息上。后面的六个字段用于安全模型，当消息处理中调用安全服务时需要。最后，PDU报文，连同contextEngineID和contextName参数，组成PDU范围，用于PDU处理。

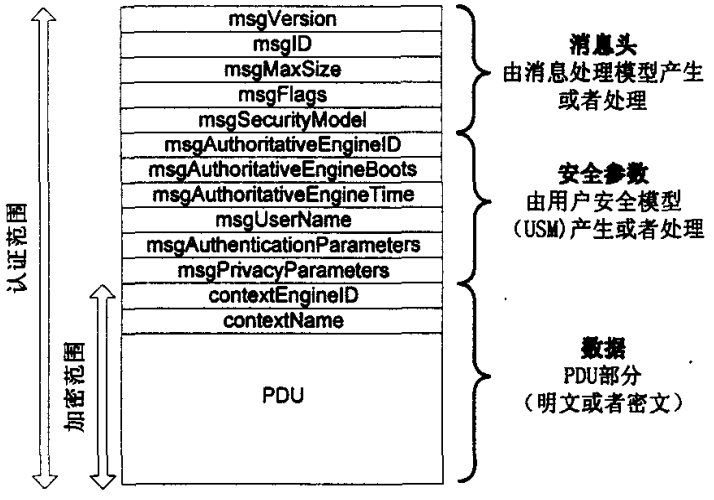

图5-2 SNMPv3中的消息结构图

最先五个字段如下：

（1）msgVersiong：填与SNMPv3相应的值3。

（2）msgID：区别两个SNMP实体的唯一标识。它可以区分消息的请求者和响应者，也被用于协调体系中不同子系统模型的消息处理。ID的范围：0~2^{31}-1。

（3）msgMaxSize：消息发送者支持的最大消息长度（8进制），范围从484~2^{31}-1。这是发送者能够从其他SNMP引擎接收的最大长度。

（4）msgFlags：一个8进制串，在三位中包括三个标记：reportableFlag，privFlag，authFlag。如果reportableFlag=1，那么在满足产生一个Report PDU的条

件时，必须返回给发送者；reportableFlag=0，可以不发送Report PDU。当消息中包含请求（如Get、Set）或者Inform时，reportableFlag被发送者设置为1；当消息中包含响应、Trap或者Report PDU时，被设置为0。ReportableFlag是用于辅助决定什么时候发送Report，即当消息的PDU部分不能被解密时（例如因为密钥错误）。发送者用privFlag和authFlag标识运用到中消息的安全级别。当privFlag=1，表示运用了加密；authFlag=1，表示运用了认证。所有的组合（除privFlag=1且authFlag=0外）都允许，即不允许加密不认证。

（5）msgSecurityModel：标识符，范围0~2^{31}-1，表示发送者准备消息时运用的安全模型，该模型也要被接收者用来处理消息。保留值1、2、3，分别对应SNMPv1、SNMPv2、SNMPv3。

当一条流出的消息由消息处理器交给USM时，USM填写消息头中相应的安全参数。当一条流入的消息由消息处理器交给USM时，USM处理相应的字段。安全相关的参数包括：

（1）msgAuthoritativeEngineID：填写该消息交换中经过认证的SNMP引擎中的参数snmpEngineID的值。因此，这个值表示命令（如Trap、Response、Report）的源和命令（如Get、GetNext、GetBulk、Set、Inform）的目的。

（2）msgAuthoritativeEngineBoots：填写该消息交换中经过认证的SNMP引擎中的参数snmpEngineBoots的值。snmpEngineBoots的范围：0~2^{31}-1，表示从初始配置后到现在的初始化或者重新初始化的次数。

（3）msgAuthoritativeEngineTime：填写该消息交换中经过认证的SNMP引擎中的参数snmpEngineTime的值。snmpEngineTime的范围：0~2^{31}-1，表示认证的SNMP引擎上次增加snmpEngineBoots后到现在的时间（秒数）。每个认证的SNMP引擎负责每秒将自己引擎的变量snmpEngineTime值增加1。

（4）msgUserName：消息以哪个用户（演员）的名义进行交换。

（5）msgAuthenticationParameters：如果消息交换中没有用到认证，则该域为空；否则，该域为一个认证参数。对于当前的USM定义，认证参数为HMAC计算出的消息认证码。

（6）msgPrivacyParameters：如果消息交换中没有用到认证，则该域为空；否则，该域为一个加密参数。对于当前的USM定义，

模型中，授权模块要提供对HMAC-MD5-96和HMAC-SHA-96的支持，加密模块提供对

CBC-DES的支持。时间模块是作为模型内嵌部分实现。USM中允许其它类似的协议作为替代或扩充。

在基于用户的安全模型中，如果用了授权认证，那么，数据完整性检查就在授权认证模块中完成。如果消息要经过授权，它要通过授权认证模块的授权检查和内置的时间模块的时间检查。

5.4 基于视图的访问控制模型（VACM）

5.4.1 VACM模型的结构

简单网络管理协议中的访问控制子系统主要职责是检查对特定对象（事例）的特定的访问（读、写等）是否被允许。这里的基于视图的访问控制模型（View-based Access Control Model，VACM）可以被其它的模型替代。

在一个SNMP实体中，处理来自SNMP实体的读或修改消息时，需要访问控制。例如，命令响应器接到并处理命令产生器的请求时，要进行访问控制。这些请求包括读或写类的PDU。当通知消息产生时，也需要访问控制。

VACM提供了一套服务，应用程序可以利用这些服务来检查访问权限。为了实现这一模型，需要将访问控制的权限和策略放到本地配置数据库（Local Configuration Datastore，LCD）中，为了实现远程管理和控制，可以通过管理信息库访问这些数据。

下面是模型中的元素的定义：

1.组（Groups）：

一个Group是0个或多个<securityModel，securityName>二元组构成的集合，二元组代表以谁的权利访问管理对象。一个group定义了属于该group的所有securityName的访问权。securityModel和 securityName组合，一般对应一个group。一个group 用groupName定义。访问控制模型假定securityName是经过授权的可信任的，不需要进一步认证。

2.安全级（securityLevel）

组中不同成员的访问权限可以被定义为不同的安全级别，例如： noAuthNoPriv，authNoPriv，和 authPriv等。

3.上下文（Contexts）

一个SNMP的context是一个SNMP实体能访问的管理信息的集合。一个管理信息项可能存在于不止一个context中。一个SNMP实体可能访问多个context。

VACM中定义了一个vacmContextTable用contextName列出了本地可用的上下文。

4.MIB 视图（Views）和视图族（View Families）

由于安全的考虑，有必要将特定用户组的访问限制在一个管理域的管理信息库子集中。为了实现这一目标，提出了通过管理信息库视图（MIB view）来访问上下文。MIB view中确定了在上下文中被管理的对象类型（或对象类型的事例）。例如，给定上下文，通常在一个MIB view中，提供了对该上下文的所有管理信息的访问。其它的MIB view提供其它信息子集的访问。所以，组在每个特定上下文中的访问权限都可以用MIB view描述。

因为被管理对象类型是通过树状结构定义的，所以MIB view是由“视图子树”集合构成，每个视图子树是被管理对象的命名树的子树。当MIB表格中的一行的每一列出现在不同子树中时，它们有相似的结构，很容易归为一个结构，这个结构称为视图子树族。

5.4.2 通过VACM实现访问控制

通过这种访问控制机制，我们可以对代理进行配置，使不同的管理者对代理的MIB访问时，只能访问到允许的层次。一个代理中的实体可以通过两种方法约束特定的管理员访问它的信息库。首先，它可以限定管理者只能访问MIB的一部分。例如，代理可以限制多数的管理员只看到MIB中与性能相关的静态参数，而只允许一个特定的管理员看到重要的配置参数。其次，可以限定对该部分MIB的操作。例如， 可以限定特定的管理员只能“读”某代理MIB中的一部分。某个代理针对每个管理员的访问控制策略必须预先配置，可以用一个表格描述各种经授权的管理员的权限。和认证授权不同，访问控制不以用户为单位，而以组为单位，一个组可以对应多个用户。通过以下步骤决定是否能够访问被请求的对象：

1.VACM检查上下文表（vacmContextTable）中是否包含要求的上下文名（contextName）。如果有，那么该上下文就是该SNMP引擎所能识别的，否则，就返回一个没有该上下文错误（noSuchContext）。

2.VACM检查安全-组对应表（vacmSecurityToGroupTable）是否包含要求的二元

组<securityModel，securityName>对应的组。如果有，那么在这种安全模型securityModel下操作的该演员就是该SNMP引擎中某一组的成员。否则，就返回一个没有该组错误（noGroup）。

3.VACM查询访问控制表（vacmAccessTable），用组名（groupName）、上下文名（contextName）、安全模型名（securityModel）和安全级别（securityLevel）作为索引。如果找到了相应的单元，说明在该安全模型、安全级别及上下文下，该组（groupName）有访问控制策略对应。否则，将返回一个没有访问单元错误（noAccessEntry）。

4.VACM将检查选定的访问控制表（vacmAccessTable）中的单元是否包含某个视图类型（viewType）（如读、写和通知）相应的MIB视图。如果有，那么这个单元将包含与组名、 上下文名、安全模型、安全级别和视图类型对应的视图名 viewName；否则将返回错误：没有该视图（noSuchView）。

5.上一步得到的视图名viewName后，可以用它作为索引，查找视图树族表（vacmViewTreeFamily）。如果找到相应的MIB视图，那么说明对该视图名viewName对应一个经过配置的视图；否则，也将返回错误：没有该视图。

6.VACM将请求的变量名variableName与选中的MIB视图树对照。如果该变量在视图中，那么将返回一个状态信息表示允许访问，否则，返回错误：不在视图中（notInView）。

如果以上几步中返回错误，则用户不能访问该MIB中的对象；如果返回允许访问，则SNMP代理将按对象类型及用户请求执行相应操作。

5.5 CORBA机制对子系统安全的帮助

除了通过基于用户的安全模型以及对MIB运用基于视图的访问控制外，我们还采用了一些策略来保证系统的安全。网络监视与管理子系统采用Java applet作为远程管理控制端接口，由管理员选择需要装载的管理信息库，通过GUI界面发出相应的管理操作命令（如读取、设置MIB变量的值，监测某一变量的变化过程等）。而所有这些操作，实际上是浏览器中执行Java applet中ORB桩代码，通过CORBA机制，调用了SNMP管理服务器上的ORB框架代码，由这些框架代码执行实际的管理命令。这种分布

式终端界面与集中管理服务器结合的方式，使得被管理的设备只对集中的管理服务器负责，减少了系统的复杂度，降低了用户误操作的可能，同时，避免了多个管理站带来的安全问题。

5.6 小结

通过基于用户的安全模型和基于视图的访问控制，结合CORBA和适当的通信加密技术，能够有效的保证网络监视与管理子系统自身的安全。

6 结合SNMP的入侵检测子系统的设计与实现

6.1 设计思想

利用SNMP获取网络设备中的数据，计算出如网络流量、重传率、误码率等参数，设定合适的阈值从而发现可疑的行为。

检测数据可以采用很多存储的方式。如文件、数据库存储或者XML格式等。我们主要以数据库的形式共享检测数据。

对于网络拓扑结构，我们将网络关键设备简化为静态树形结构处理，采用数据库分层存放，没有采用复杂的动态生成方法。在一些文献中有专门的关于网络拓扑自动发现的论述[49, 50]。

6.2 数据库设计

被管理设备基本信息表的表名 main，表结构如表6-1所示。

表6-1 被管理设备基本信息表

字段名	类型	含义	数据示例
NAME	CHAR(30)	设备IP地址	192.168.1.1
CODE	CHAR(10)	设备编码	1
COLOR	CHAR(6)	设备颜色（表示状态）	GREEN
DEVICE	CHAR(10)	设备类型	MDR
ADDRESS	CHAR(10)	设备位置	南一楼510
CHECKUTILITY	CHAR(5)	检查标记（YES：检查）	NO
TIMER	CHAR(3)	检查时间间隔(分钟)	5
HISTORY	CHAR(4)	历史记录标记（YES：有历史记录）	NO

被管理设备MIB对象名维护表的表名为mdr，表结构如表6-2所示。

表6-2 被管理设备MIB对象名维护表

字段名	类型	含义	数据示例
NAME	CHAR(30)	Mib对象名	SysUpTime
MIBVALUE	CHAR(40)	对象OID标识	.1.3.6.1.2.1.1.3
STATE	CHAR(10)	对象状态	Enable
ACCESS	CHAR(15)	访问许可	Readonly
SYNTAX	CHAR(15)	访问标志	NORMALE

被管理设备常用MIB对象值表的表名为mainMDR，表结构如表6-3所示。

表6-3 被管理设备常用MIB对象值表

字段名	类型	含义
Name	CHAR(30)	设备IP地址
SysUpTime	CHAR(50)	系统运行后时间
SysName	CHAR(50)	设备名称
SysContact	CHAR(50)	设备联系方式
SysLoaction	CHAR(50)	设备位置
ProductMacAddr	CHAR(20)	产品Mac地址
Gate	CHAR(20)	网关
Mask	CHAR(20)	掩码
BeaconsSuccess	CHAR(10)	成功数据包数
BeaconsMissed	CHAR(10)	失败数据包数

为了节省监测工作的时间，将实际中的树行网络拓扑结构以多个表的形式分层存放。如果上一级的设备断开就不用测试其对应的子节点设备。这里，每个表名与节点的IP地址对应，如果节点IP地址为aaa.bbb.ccc.ddd，那么表名的表达式如下：

f(aaa.bbb.ccc.ddd)="ip"+aaa+"A"+bbb+"A"+ccc+"A"+ddd

其中，运算+为字符的连接运算。例如IP地址为192.168.1.1的子节点对应的表名为ip192A168A1A1 。

拓扑结构分层存放的表结构如表6-4所示。

表6-4 拓扑结构分层存放表

字段名	类型	含义	数据示例
Linknod	CHAR(15)	子节点IP	192.168.1.2
Num	INT	子节点的儿子数	2

例如，某网络拓扑结构示意图如图6-1所示。

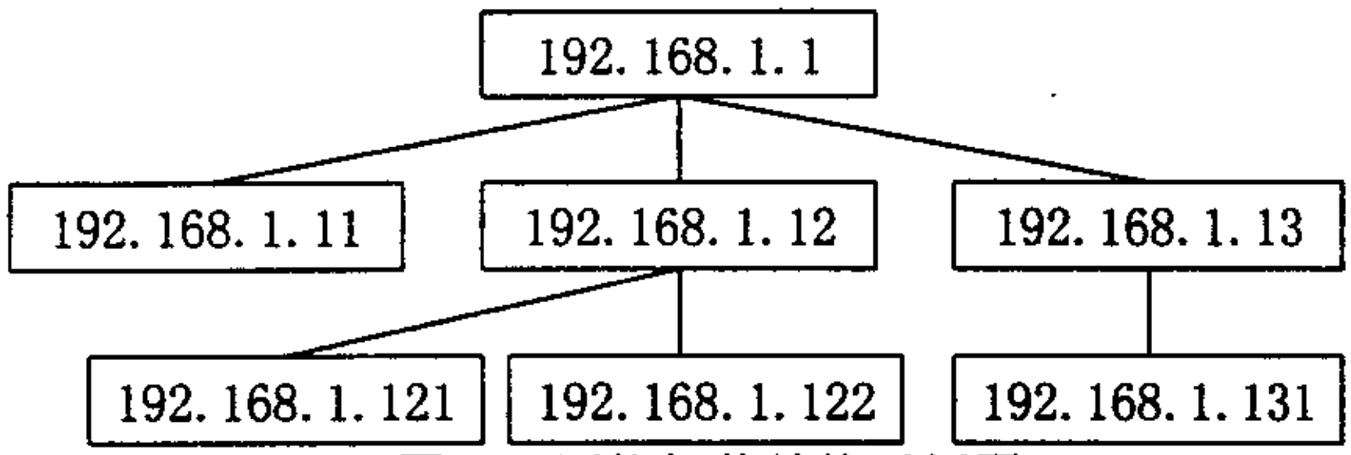

图6-1 网络拓扑结构示例图

该拓扑结构可以用表6-5、表6-6和表6-7共同表示。

表6-5 分层次存放网络拓扑结构表ip192A168A1A1

192.168.1.11	0
192.168.1.12	2
192.168.1.13	1

表6-6 分层次存放网络拓扑结构表ip192A168A1A12

192.168.1.121	0
192.168.1.122	0

表6-7 分层次存放网络拓扑结构表ip192A168A1A13

192.168.1.131	0

其中，叶子节点IP地址对应的num=0，表示没有后继节点，所以地址192.168.1.11、192.168.1.121、192.168.1.122、192.168.1.131都不需要与之对应的表。

节点主要被检测参数表中，表名对应表达式为

U(aaa.bbb.ccc.ddd)="UTI"+aaa+"A"+bbb+"A"+ccc+"A"+ddd

被检测参数表中，被检测参数的具体内容与实际设备参数相关。

6.3 检测采用的机制

1. 管理设备变动相对较少的参数（如联系人、地址、设备网关、掩码）或者有参考价值的数据（如系统运行时间、最大传输速度、信道利用率），用数据库进行记录，定期检查，可以观察系统的运行情况，作为入侵检测的参考。

2. 网络拓扑结构是在检测子系统启动时读取，存放在内存中，以加快系统运行的速度。

3. 管理员可启动对某些关键节点设备的检测功能，在控制中心的服务程序将检测这些节点中关键参数，当关键参数满足定义的阈值时，将数据录入数据库。管理员或者有相应权限的用户可以查询这些记录。这种检测采用的是一种轮询方式，即通过访问网络拓扑结构转化成的层次表，遍历整个网络，直至没有后继节点或者节点状态为不可用。在轮询过程中，对CheckUtility='YES'的节点（由管理员配置）进行常用MIB对象检测，检测的时间间隔为Timer。将相应的参数经过计算，得到相关信息。如：链路发送流量、平均传输速率、QPSK方式比例、QAM 方式比例、带宽利用率、丢包率、QPSK重传率、QAM 重传率、链路重传率、链路接收流量、链路总流量、链路利用率。主要流程如图6-2：

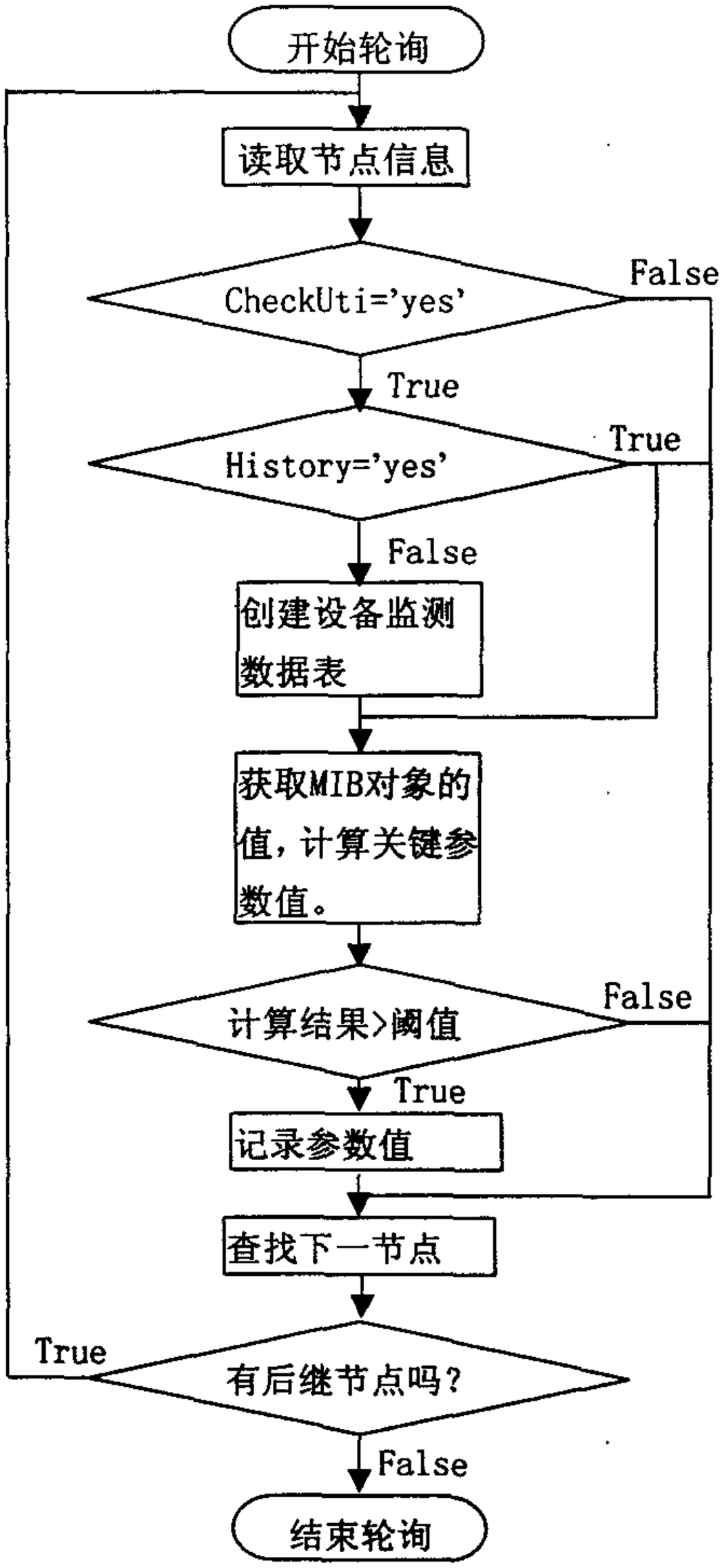
开始轮询
读取节点信息
CheckUti='yes'
False
True
History='yes'
True
False
创建设备监测数据表
获取MIB对象的值，计算关键参数值。
计算结果>阈值
False
True
记录参数值
查找下一节点
有后继节点吗？
True
False
结束轮询

图6-2 轮循检测流程图

6.4 小结

利用简单网络管理协议，对管理信息库中的某些对象进行监测，通过一定计算得到与网络相关的参数，对入侵检测系统有很大的参考价值。

7 总结

本课题组在通用入侵检测系统框架模型基础上，提出的分布式基于代理的网络入侵检测系统主要具有如下特点：

1. 基于CIDF模型的体系结构，使各模块有独立性；
2. 探测和分析代理可进行并行处理，从而提高了检测系统的效率；
3. 基于协议的分析检测提高了系统检测的准确率。

本文的重点是将入侵检测与网络管理结合，这一思想是入侵检测系统研究中的创新。通过无线网络管理方面的工作经验，在对现有的入侵检测系统进行研究后，发现入侵检测和网络管理有很多共同点。首先，两者有共同的目标：监视、管理计算机网络资源，使这些资源安全、正常的为合法用户服务。其次，两者在系统结构上有类似之处，都采用了分布式结构。本文在分布式基于代理的入侵检测系统的基础上，设计并实现了网络管理子系统，该子系统在以下的方面对原有的入侵检测系统有重大改进：

1. 提高了入侵检测的准确度

结合标准的简单网络管理协议SNMP，入侵检测系统可以获取一般方法无法获取的网络设备信息，利用这些高层信息，通过进一步的分析处理，可以降低入侵检测系统的误报率，使系统工作得更加稳定。

2. 增加了防范的手段

当发现入侵时，系统可以利用SNMP协议，对不同的网络设备（如路由器、防火墙、工作站）进行调整，例如重启某个被入侵利用的路由器链路、修改防火墙过滤规则或者暂时停止某些服务等。

3. 提高了系统的协同工作能力

不同的入侵检测系统如何协同工作也是长期入侵检测研究的工作之一，而本文中采用简单网络协议进行通信和管理，在通信代理中实现对SNMP的支持，使系统具有良好的可扩展性和协同工作的能力。通过增加合适的MIB对象及相应操作，利用通用的网络管理手段，还可以了解到入侵检测系统自身的工作状态，并能进行系统的配置工作。

4. 提高了系统自身的安全

SNMPv3中有比较完善的安全解决方案。适当运用SNMPv3的安全功能，结合其他安全措施，可以在一定程度上保证入侵检测系统本身的安全。

5. 提供了方便的远程管理功能

将WWW服务、Java Applet与系统结合，形成方便的远程管理模式，管理员只需要通过浏览器，在任意的主机上下载并运行Java小程序，就可以对整个网络系统进行监视和控制，而不用安装额外的程序。同时，利用CORBA技术，减少了分布式管理的安全隐患。在进行远程管理操作时，系统还允许管理员方便的加载不同厂家设备相应的管理信息库，从而实现了对多个不同设备的兼容管理。

我们的入侵检测系统也存在一些问题，这也是我们下一步需要研究的工作。

1. 提高系统效率

我们只实现了对几种典型入侵的检测，因为对截取报文的分析消耗一定的资源，检测类型越多，消耗的资源越多，如何提高检测效率，支持更多检测类型是今后研究的方向之一。另外，Java Applet的执行速度比较慢，也需要进一步解决。

2. 分布式远程管理引发的问题

在网络监视和管理子系统中，为了实现远程分布式管理，我们用了浏览器、Java Applet模式。但Java Applet有自身的安全措施，很多操作不能直接实现，比如将日志信息写入存储器中等。所以我们引入了简单的CORBA机制，但如何设计更加合理、高效的ORB对象还需要继续研究。

致谢

本文是在我的导师余祥宣教授的指导下完成的。余老师在学习和生活上都给了我很大的帮助和鼓励。同时，洪帆教授对本文提出了很多宝贵的意见。余老师、洪老师严谨求实、对科学前沿敏锐的洞察力、勇往无前的开拓精神以及对我的严格要求都使我收益匪浅。

课题组内的崔国华教授、胡伦骏副教授、付小青副教授，长期以来一直关心我的成长，为我创建了良好的学习、工作环境，给予了我悉心的指导和教育，在此对他们表示感谢。

感谢汤学明、吴辉军博士、杜小勇、张明、杨洋、文珠穆、裴鹏军、彭琨、谭谦仁、姚勇、度燕、李孟珂等师兄师姐，以及刘建农、卢刚、刘刚、何中华、黄正波、崔永泉、邓集波、刘伟、李静、陈凤珍、赵晓斐、付才、卢涛、吴敏等同学。完成任何工作都需要大家的团结合作，在思想的交流中才能碰撞出智慧的火花，真心感谢所有关心、帮助和支持过我的人。

参考文献

[1] 王锡林，郭庆平等. 计算机安全. 人民邮电出版社，1995年1月.21～55

[2] Computer Emergency Response Team, Pittsburgh. IP Spoofing Attacks and Hijacked Terminal Connections. Computer Networks, January 1995,3: 2～9

[3] Woolf S. Woodcock, B. Operator Requirements of Infrastructure Management Methods. Information Security Bulletin, 2001,2: 22～35

[4] Amoroso, Edward. Intrusion Detection. Intrusion.net Books. Sparta, NewJersey, 1999. 33～64

[5] Amoroso, Kwapniewski R. A selection criteria for intrusion detection system. Computer Security Applications Conf, 1998,14: 280～288

[6] Denning, Dorothy E. An Intrusion Detection Model. IEEE Transactions on Software Engineering, February 1987, Vol. SE-13, No. 2: 222～232

[7] Boeckman C. Getting closer to policy-based intrusion detection. Information Security Bulletin, 2000,5(4): 13～22

[8] Mark Crosbie and E. H. Spafford. Active Defense of a Computer System Using Autonomous Agents. Department of Computer Sciences, Purdue University, 1995,2: 18～24

[9] David L. Tennenhouse, Jonathan M. Smith, W. David Sincoskie, David J. Wetherall, and Gary J. Minden. A Survey of Active Network Research. IEEE Communications Magazine, January 1997, Vol. 35, No. 1: 80～86

[10] Lippmann R, Fried D, Graf I, et al. Evaluating intrusion detection systems: The 1998 DAPA Offline Intrusion Detection Evaluation. Discex, 2002,2: 12～26

[11] Durst R, Champion T, Witten B, et al. Testing and evaluating computer intrusion detection systems. Comm. ACM, 1999, 42(7): 53～61

[12] Vern P. Bro: A system for detecting network intruders in real-time.

Computer Networks, 1999, 31: 2435～2463

[13] M. Asaka, S. Okazawa, A. Taguchi, and S. Goto. A Method of Tracing Intruders by Use of Mobile Agents. INET'99, June 1999, 2: 13～34

[14] B. Barak, A. Herzberg, D. Naor, E. Shai. The proactive security toolkit and applications. 6th ACM Conference on Computer and Communications Security, 1999, 43(8): 34～68

[15] Ning P, Wang X S, jajodia S. Modeling requests among cooperating intrusion detection systems. Computer Communications, 2000, 23: 1702～1715

[16] Firth, Robert. Detecting Signs of Intrusion. Pittsburgh. Software Engineering Institute. Carnegie Mellon University, 1997, 10(7): 25～43

[17] Huang Mingyuh, Robert J J. Thomas M W. A large scale distributed intrusion detection framework based on attack strategy analysis. Computer Networks, 1999, 31: 2465～2475

[18] Eugene H, Diego Z. Intrusion detection using autonomous agents. Computer Networks, 2000, 34: 547～570

[19] Stevens W R. TCP/IP Illustrated Vol 1: The Protocols. USA: Addison-Wesley Reading Mass, 1994. 170～197

[20] Terry Escamilla. Intrusion Detection: Network Security Beyond the Firewall. Computer Networks, 1998. 225～260

[21] 余祥宣，刘铭，检测、防范DoS攻击的分布式模型及实现. 华中科技大学学报（自然科学版），2002年3月，vol.30，No.3，19～21

[22] Kochmar, John. Preparing to Detect Signs of Intrusion. Pittsburgh. PA: Software Engineering Institute, Carnegie Mellon University, 1998, 8: 44～60

[23] Northcutt, Stephen. Computer Security Incident Handling: Step-by-Step. The Fourth Annual UNIX and NT Network Security Conference. Orlando, FL: The SANS Institute, October 4-31, 1998. 80～124

[24] S. Kent and R. Atkinson. Security Architecture for the Internet Protocol. Network Working Group, Request for Comments 2401, November 1998. 48～73

[25] Edward G. Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Traps, Trace Back, and Response. Intrusion Net Books, 1999, 12～34

[26] Roesch, Martin. Snort - Lightweight Intrusion Detection for Networks. USENIX LISA Conference, November 1999. 43～59

[27] Staniford, S., J. Hoagland and J. McAlerney. Practical Automated Detection of Stealthy Portscans. ACM CCS IDS Workshop, November 2000. 134～165

[28] R. Hauser, T. Przygienda, and G. Tsudik. Reducing the Cost of Security in Link-State Routing. Internet Society Symposium on Network and Distributed Systems Security, 1997. 23～48

[29] Reilly, Mark. A Scalable Intrusion Detection and Response - Framework System Specification. ORA Technical Report TM-97-0032, December 1997. 78～93

[30] L.T. Heberlein, G.Dias, K.Levitt, B.Mukherjee, J.Wood, and D. Wolber. A Network Security Monitor. Symposium on Research in Security and Privacy, May 1990, 296～304

[31] Wayne Jansen and Tom Karygiannis. Mobile Agent Security. National *Institutes of Standards and Technology*, NIST SP 800-19, August 1999. 28～52

[32] Mukherjee B, Heberlein L T, Levitt K N. Network Intrusion Detecion. IEEE Network, 1994, 8(3): 26～41

[33] Gregory B. White, Eric A. Fisch, and Udo W. Pooch. Cooperating Security Managers: A peer-based intrusion detection system. IEEE Network, January 1996, 10(1), 20～23,

[34] 陈晓苏，宁翔，肖道举. 一种基于CIDF的入侵检测系统模型. 华中科技大学学报（自然科学版），2002年3月，Vol.30, No.3, 1～3

[35] 陈晓苏，姜朝，肖道举. 基于高性能网络的入侵检测系统架构. 华中科技大学学报（自然科学版），2002年3月，Vol.30, No.3, 4～6

[36] Subramanian, M. Network Management, Principles and Practice.

Addison-Wesley, 1995,230～243

[37] Harrison, C.G., D.M. Chess, A. Kershenbaum. Mobile Agents: Are they a good idea?. IBM Research Report, March 1995,10(2): 38～64

[38] Lunt T F. A Survey of Intrusion Detection Techniques. Computers and Security,1993,12: 405～418

[39] Ravi S. Sandhu. Transaction control expressions for separation of duties. In Fourth Annual Computer Security Application Conference, Orlando, FL, December 1988,282～286

[40] Jacobs, S., D. Dumas, W. Booth, M. Little. Security Architecture for Intelligent Agent Based Vulnerability Analysis. Proceedings: 3rd Annual Fedlab Symposium on Advanced Telecommunications/Information Distribution Research Program, Orlando, FL, February 1999. 447～451

[41] 张明. 数据库与网络通信安全的封装与隔离研究——通信代理的设计与实现: [硕士学位论文]。武汉: 华中科技大学图书馆, 2000年

[42] 余祥宣、徐智勇, 何绪斌. 网络环境下的身份验证. 计算机与数字工程, 1996, 24(5): 45～47

[43] 裴鹏军. 分布式网络环境下的认证和密钥管理: [硕士学位论文]。武汉: 华中科技大学图书馆, 2000年

[44] 彭昆. 隐通道的分析与测试: [硕士学位论文]。武汉: 华中科技大学图书馆, 2000年

[45] Lyndon G. Pierson. Integrating End-to-End Encryption and Authentication Technology into Broadband Networks. Sandia National Laboratories, March 1996. 69～90

[46] John F. Barkley, Anthony V. Clincotta, David F. Ferraiolo, Serban Gavrilla and D. Richard Kuhn. Role Based Access Control For the World Wide Web. National Institute of Standards and Technolog, Gaithersburg, Maryland, April 8,1997. 77～89

[47] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman. Role-based access control models. IEEE Computer, February 1996,29(2): 23～51,

[48] Buerger, D.J. Virtual LAN cost savings will stay virtual until networking's next era. Network World, March 1995,3(1): 121～145

[49] Passmore, D., Freeman, J. The Virtual LAN Technology Report. Network World, March 1997,4(2): 33～59

[50] Hein, M. Griffiths, D., Berry, O. Switching Technology in the Loacl Network: From LAN to Switched LAN to Virtual LAN. Thompson Computer Press, February 1997. 79～90

附录1　攻读学位期间发表论文目录

[1] 余祥宣，刘铭，检测、防范DoS攻击的分布式模型及实现，华中科技大学学报(自然科学版)，2002年3月，vol.30，No.3，19～21

附录2 附图列表

附录3 附表列表