摘要

入侵检测在计算机安全系统中发挥着越来越重要的作用，目前入侵检测使用的规则库还是主要依赖于专家分析提取，由于入侵检测系统中数据量很大，使用人工分析的代价是昂贵的。用数据挖掘技术分析网络数据进行入侵检测，可以有效的减少人工分析的工作量，但数据挖掘技术应用到入侵检测中存在着一些问题，特别是挖掘效率的问题，该文通过改进系统结构和数据挖掘的算法来提高系统的性能。

本文对入侵检测技术进行了分析和研究，提出了一个基于数据挖掘和 Agent 的分布式入侵检测模型，并在此模型的基础上实现了一个基于数据挖掘和 Agent 的分布式入侵检测系统。该系统可以分布在网络中任意数目的主机上，其分布式体系结构和灵活的代理体系，使得系统具有很强的分布性和扩展性，其实时的学习功能增强了入侵检测系统的检测能力。

基于数据挖掘和 Agent 的分布式入侵检测模型的代理体系主要由通信代理、数据挖掘代理、入侵检测代理组成，代理之间各自独立又相互协作，合作完成入侵检测的任务。代理体系的引入，将基于主机的入侵检测和基于网络的入侵检测灵活地融为一体，形成一个统一的入侵检测系统。数据挖掘技术的引入，使入侵检测系统有了自学习能力，这使系统能自己从本机日志或网络上的数据中学习并提取特征值，通过分析不断扩充自己的规则库，不断提高检测入侵的能力。

本文实现了基于数据挖掘和 Agent 的分布式入侵检测系统的具体功能，本文后面章节对引用的数据挖掘的聚类算法和关联规则进行了详细的分析，并通过数据实验证明其可行性，系统可以运行在 Unix/Linux 主机上，经测试证明是一个可行而且先进的入侵检测系统。

关键词 入侵检测；数据挖掘；聚类分析；关联规则；分布式代理；

Abstract

Intrusion detection plays a more and more important role in computer security. In conventional way, experts analyze data collected by intrusion detection system and abstract rule sets. Manual analysis is quite expensive because of enormous amount of data. Applying data mining technique in intrusion detection can reduce workload of manual analysis. However data mining technique also has some problems, such as efficiency and accuracy. This article proposes system architecture and datamining algorithm to enhance the efficiency of datamining intrusion detection system.

Intrusion detection technology is analyzed and explored in this thesis, and a Datamining and Agent-based Distributed Intrusion Detection Model is presented in this thesis. Based on this model, a Datamining and Agent-based Distributed Intrusion Detection System called DA_DIDS is developed. DA_DIDS can be distributed on any host in the network. DA_DIDS has good distributed and scalable ability because of its distributed architecture and flexible agent system. DA_DIDS' s real-time study ability can enhance the system' s detection effect.

The agent system of DA_DIDS consists of three main modules: transport service agent, datamining agent, intrusion detection agent. Every agent works independently and together to accomplish intrusion detection. Due to the agent system, DA_DIDS comes into being a uniform intrusion detection system that is competent for the task of host-based and network-based intrusion detection. The introduction of the datamining technique gives the study ability to intrusion detection system, this ability makes the system to study and extract eigenvalues from logs of the host or from the network, then enlarge DA_DIDS' s rule database, the ability to detect intrusion can be enforced ceaselessly.

In this thesis, we implement the concrete functions of DA_DIDS, clustering algorithm of datamining is analyzed detailedly in the posterior chapters and is proved feasible through the experimentation. DA_DIDS has be implemented and run on UNIX/LINUX system. It shows that this is an advanced intrusion detection system.

Keywords Intrusion Detection; Datamining; Cluster Analysis; Association Rules; Distributed Agent;

独 创 性 声 明

本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，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，也不包含为获得北京工业大学或其它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了谢意。

签名：[illegible] 日期：2004.6.8

关于论文使用授权的说明

本人完全了解北京工业大学有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。

（保密的论文在解密后应遵守此规定）

签名：[illegible] 导师签名：[illegible] 日期：2004.6.8

第1章 绪论

1.1 课题背景

在计算机网络中，单纯凭借各种密码学的方法无法达到好的安全效果，入侵检测越来越显示出它的重要性。随着互联网在我国的普及和网络应用的深入，各级政府部门，教育和科研机构，企事业单位，乃至个人组织都相继推出了计算机网络服务，从根本上改变了人们的生活和工作方式。人们在提供网络服务、参与网络活动的同时，往往只看到其便利、有益的一面，而降低了警惕性，忽视了潜在于网络中的安全问题。以往的教训表明，如果对网络安全问题掉以轻心，互联网络也可能会使服务提供者的形象严重受损，带来重大经济损失，甚至造成恶劣的社会影响。互联网具有天然的开放性和协议的简便性，它在展示其无所不能的强大威力的同时，也不可避免地伴随了大量的安全隐患。网络结构组织各方面的缺陷，系统与应用软件的漏洞，以及网络管理员的水平低下和疏忽大意，都可能使网络攻击者有机可乘；恶意的入侵者干扰正常业务，销毁或篡改重要数据，甚至使更多的服务器失去控制，造成一系列严重后果。

人们在得益于信息革命所带来的新的巨大机遇的同时，也不得不面对信息安全问题的严峻考验。伴随着对网络技术的推进，网络攻防的战斗也越演越烈。网络安全已经引起各国、各部门、各行各业的高度重视，在网络中引入安全防范机制已经成为人们的共识。防范网络攻击，最常用的对策是构建防火墙。防火墙是一种应用层网关，按照设定的规则对进入网络的 IP 分组进行过滤，同时也能针对各种网络应用提供相应的安全服务。利用防火墙技术，经过仔细的配置，通常能在内外网之间实施安全的网络保护机制，降低风险。但仅仅使用防火墙保障网络安全是远远不够的，因为入侵者会想方设法寻找防火墙背后可能敞开的通道。另一方面，防火墙在阻止内部袭击等方面也收效甚微，对于企业内部心怀不满而又技术高超的员工来说，防火墙形同虚设。而且，由于性能的限制，防火墙通常不能提供有效的入侵检测能力。因此，认为在 Internet 入口处部署防火墙系统就足够安全的想法是不切实际的。

能否成功阻止网络黑客的入侵，保障计算机和网络系统的安全和正常运行，已经成为各机构和单位能否正常运作的关键性问题。入侵检测系统 IDS 是近年出现的新型网络安全技术，是一套软件和硬件的结合体。IDS 能弥补防火墙的不足，为受保护网络提供有效的入侵检测及采取相应的防护手段；入侵检测是一个全新的、迅速发展的领域，并且已成为网络安全中极为重要的一个课题；入侵检测的方法和产品也在不断的研究和开发之中，并且已经在网络攻防实例中初步展现出其重要价值。

随着网络信息的快速增长和存储信息的无限扩大，怎样对网络进行监控，将面临相当大的困难。虽然我们拥有更为高性能的计算机，但这远远赶不上所要分析的网络数据的要求。同时如何实时地分析和处理网络信息，将是我们所面临的一大挑战。下一代的网络入侵检测技术将致力于开创新的技术，让入侵检测系统能适应高带宽和高负荷的网络环境，支持最新的计算机网络协议，并有自我学习的能力。这其中应用比较突出的是数据挖掘和知识发现(Knowledge Discovery and Data Mining，DMKD)[1]技术，该技术的研究正蓬勃发展，越来越显示出其强

大的生命。

1.2 数据挖掘技术

数据库技术的成熟和数据应用的普及使得人类积累的数据量正在以指数速度增长。数据洪水正向人们滚滚涌来。例如，在超级市场通过条型码扫描，把每一笔商品交易输入数据库，一个中型超市经营的商品就数万种。如此大量的数据在传统的数据库中并不能很好地回答经理关心的问题：商品在不同季节或一天的不同时间的销售量有何变化规律？商品 A 的销售量的增加是否会同时带动商品 B 的销售？如何调整商品的资金占用，以达到最佳的资源配置？各种商品的销售之间是否存在一定的关联关系？

大量信息在给人们带来方便的同时也带来了一大堆问题：第一是信息过量，难以消化；第二是信息真假难以辨识；第三是信息安全难以保证；第四是信息形式不一致；难以统一处理。人们开始提出一个新的口号："要学会抛弃信息"，并且开始考虑如何才能不被信息淹没，而是从中及时发现有用的知识、提高信息利用率。

另一方面，随着大量的大规模的数据库迅速不断地增长，人们对数据库的应用已不满足于仅对数据库进行查询和检索，仅用查询检索不能帮助用户从数据中提取带有结论性的有用信息。这样数据库中蕴藏的丰富知识，就得不到充分的发掘和利用，从而造成了信息的浪费，由此也会产生大量的数据垃圾。从人工智能应用来看，专家系统的研究虽然取得了一定的进展。但是，知识获取仍然是专家系统研究中的瓶颈。知识工程师从领域专家处获取知识是非常复杂的个人到个人之间的交互过程，具有很强的个性，没有统一的办法。因此有必要考虑从数据库中自动挖掘新的知识。面对这一挑战，数据库中的知识发现(Knowledge Discovery Database，KDD)技术应运而生，并显示出强大的生命力。KDD 的核心是数据挖掘(Data Mining)[2]技术，实际的使用中人们往往不严格区分数据挖掘和数据库中的知识发现，把两者混淆使用。一般在科研领域中称为 KDD，而在，工程领域则称为数据挖掘。

数据挖掘技术出现于 20 世纪 80 年代后期，90 年代有了突飞猛进的发展。数据挖掘技术是人们长期对数据库技术进行研究和开发的结果。起初各种商业数据是存储在计算机的数据库中的，然后发展到可对数据库进行查询和访问，进而发展到对数据库的即时遍历。数据挖掘使数据库技术进入了一个更高级的阶段，它不仅能对过去的数据进行查询和遍历，并且能够找出过去数据之间的潜在联系，从而促进信息的传递。现在数据挖掘技术在商业应用中已经可以马上投入使用，因为对这种技术进行支持的三种基础技术已经发展成熟，他们是：

- 海量数据搜集
- 强大的多处理器计算机
- 数据挖掘算法

数据挖掘的核心模块技术历经了数十年的发展，其中包括数理统计、人工智能、机器学习[3]。今天，这些成熟的技术，加上高性能的关系数据库引擎以及广泛的数据集成，让数据挖掘技术在当前的数据仓库环境中进入了实用的阶段。

数据挖掘技术从一开始就是面向应用的。它不仅是面向特定数据库的简单检索查询调用，而且要对这些数据进行微观、中观乃至宏观的统计、分析、综合和推理，以指导实际问题的求解，企图发现事件间的相互关联，甚至利用已有的数

据对未来的活动进行预测。例如加拿大 BC 省电话公司要求加拿大 Simon Fraser 大学 KDD 研究组，根据其拥有十多年的客户数据，总结、分析并提出新的电话收费和管理办法，制定既有利于公司又有利于客户的优惠政策。美国著名国家篮球队 NBA 的教练，利用 IBM 公司提供的数据挖掘技术，临场决定替换队员，一度在数据库界被传为佳话。这样一来，就把人们对数据的应用，从低层次的末端查询操作，提高到为各级经营决策者提供决策支持。这种需求驱动力，比数据库查询更为强大。同时需要指出的是，这里所说的知识发现，不是要求发现放之四海而皆准的真理，也不是要去发现崭新的自然科学定理和纯数学公式，更不是什么机器定理证明。所有发现的知识都是相对的，是有特定前提和约束条件、面向特定领域的，同时还要能够易于被用户理解，最好能用自然语言表达发现结果。因此 DMKD 的研究成果很讲求实际。

1.3 国内外发展现状

根据调查结果，将数据挖掘应用于入侵检测已经成为一个研究热点，在这个领域已经有了很多篇论文。但是真正实现这样一套系统的还不多见，主要是 Columbia University 的 Wenke Lee 研究组和 University of New Mexico (UNM) 的 Stephanie Forrest 研究组。国内这方面的研究则刚刚起步，中国科学院的国家信息安全重点实验室、东北大学国家软件工程研究中心等走在前列。

1、Wenke Lee 研究组

Columbia University 的 Wenke Lee 研究组[12, 19, 24, 26, 29, 30]在 1998 年参加了由美国国防部高级研究计划署(DARPA)资助的 Intrusion Detection Evaluation 计划，这次测试由 MIT 的 Lincoln 实验室提供了模拟军事网络环境中所记录的 7 周的网络流和主机系统调用记录日志，这些数据全部采用 tcpdump 和 Solaris BSM Audit Data 的格式，包括了大约 500 万次会话，其中包含上百种攻击.这些攻击分为下面 4 种主要类型：

1、拒绝服务攻击(DOS)，如 Ping of Death, Teardrop, Smurf, SYN Flood 等等；

2、远程攻击(R2L)，如基于字典的口令猜测；

3、本地用户非法提升权限的攻击(U2R)，如各种各样的缓冲区溢出攻击；

4、扫描(PROBING)，包括端口扫描和漏洞扫描。

Wenke Lee 研究组分别从网络和主机两方面进行了审计数据的挖掘处理。针对网络数据，Wenke Lee 的主要做法是使用网络服务端口(service)作为网络连接记录的类型标识，根据人最的正常记录生成各个服务类型的分类模型，在测试过程中，根据分类模型对当前的连接记录进行分类，并与实际服务类型进行比较，从而判断出该分类模型的准确性。针对主机数据，Wenke Lee 则使用了一种快速的规则学习算法 RIPPER，通过对正常调用序列的学习来预测随后发生的系统调用序列，并对结果进行了进一步的抽象分析，以降低算法的预测误差。根据 DARPA 报告，由 Columbia University 实现的基于数据挖掘的入侵检测系统在检测拒绝服务攻击和扫描方面优于其它系统，在检测本地用户非法提升权限方面与其它系统大致持平，在检测远程攻击方面，所有的系统表现都不令人满意。检测率都在 70%以下。

2、Stephanie Forrest 研究组

University of New Mexico (UNM)的 Stephanie Forrest 研究组进行的是针

对土机系统调用的审计数据分析处理，最初的思想是基于生物免疫系统的概念。无论是针对生物机体还是针对计算机系统，免疫系统的关键问题在于：使用一组稳定的、并且在不同个体之间存在足够差异的特征(features)来描述自我，从而使系统具备识别“自我/非自我”的能力。

然而，对于计算机系统来说，要解决这个问题相当困难。第一，恶意代码隐藏在正常代码之中难以区分：第二，系统可能的状态几乎是无限的，寻找一组稳定的特征来定义自我并不容易。Stephanie Forrest 使用短序列匹配算法对特定的特权程序所产生的系统调用序列进行了细致的分析，在这一领域做出了大量开创性工作。

在这之后，UNM 的另一个研究小组使用了有限自动机(FSM)来构建系统调用的描述语言，但是这种方法的效率和实用性都很差。Iowa State University 的一个小组实现了一种描述语言 Auditing Specification Language (ASL)，以描述程序的正常行为。另外，还有其它一些研究者采用了神经网络等人工智能的办法。

实验和测试结果表明，将数据挖掘技术应用于入侵检测在理论上是可行的，在技术上建立这样一套系统是可能的。其技术难点主要在于如何根据具体应用的要求，从我们关于安全的先验知识出发，提取出可以有效地反映系统特性的特殊属性，应用合适的算法进行挖掘。技术难点还在于结果的可视化以及如何将挖掘结果自动地应用到实际的入侵检测系统中。目前，国际上在这个方面的研究非常活跃，这些研究得到了美国国防部高级研究计划署(DARPA)、国家自然科学基金(NSF)的支持，可见，美国官方对此十分重视。但是，这方面的研究，包括整个将数据挖掘技术运用于入侵检测的研究，总体上还处于理论探讨阶段，离实际应用似乎还有相当的距离。

我们的工作大量参照 Winke Lee, Stephanie Forrest 以及其它研究人员的做法，并在某些方面提出改进意见和相应的算法。

1.4 基于数据挖掘的入侵检测系统

入侵检测系统[4]是检测企图破坏计算机资源的完整性、真实性和可用性的行为的软件。入侵检测技术是一种利用入侵者留下的痕迹(如试图登录的失败记录)等信息有效地发现来自外部或内部的非法入侵的技术，它以探测与控制为技术本质，发挥主动防御的作用，是网络安全中极其重要的部分。

入侵检测系统的任务就是在提取到的庞大的检测数据中找到入侵的痕迹。入侵分析过程需要将提取到的事件与入侵检测规则进行比较，从而发现入侵行为。一方面入侵检测系统需要尽可能多地提取数据以获得足够的入侵证据，而另一方面由于入侵行为的千变万化而导致判定入侵的规则越来越复杂，为了保证入侵检测的效率和满足实时性的要求，入侵分析必须在系统的性能和检测能力之间进行权衡，合理地设计分析策略，并且可能要牺牲一部分检测能力来保证系统可靠、稳定地运行并具有较快的响应速度。

分析策略是入侵分析的核心，系统检测能力很大程度上取决于分析策略。在实现上，分析策略通常定义为一些完全独立的检测规则。基于网络的入侵检测系统通常使用报文的模式匹配或模式匹配序列来定义规则，检测时将监听到的报文与模式匹配序列进行比较，根据比较的结果来判断是否有非正常的网络行为。这样以来，一个入侵行为能不能被检测出来主要就看该入侵行为的过程或其关键特

征能不能映射到基于网络报文的匹配模式序列上去。有的入侵行为很容易映射，如 ARP 欺骗[8]，但有的入侵行为是很难映射的，如从网络上下载病毒。对于有的入侵行为，即使理论上可以进行映射，但是在实现上是不可行的，比如说有的网络行为需要经过非常复杂的步骤或较长的过程才能表现其入侵特性[9]，这样的行为由于具有非常庞大的模式匹配序列，需要综合大量的数据报文来进行匹配，因而在实际上是不可行的。而有的入侵行为由于需要进行多层协议分析或有较强的上下文关系，需要消耗大量的处理能力来进行检测，因而在实现上也有很大的难度。

目前大多网络入侵检测系统的规则库都是通过手工定制的方式建立起来的，尤其是用于识别判断入侵行为的检测知识，都是由领域专家自己总结提供，并将其编写到这些网络入侵检测系统中的。这类入侵检测系统存在的一个最大不足就是：它需要由人类专家不断总结提供有关的入侵检测知识。这也就意味着：它只能被动依靠外界提供的检测知识，来帮助发现网络系统的非法入侵行为，而很难发现未知的入侵或攻击行为。

数据挖掘技术在入侵检测系统中的应用尽量减少了手工和经验的成分。基于数据挖掘的网络入侵检测模型可以进行机器学习和模式扩充，入侵检测效率和可靠性明显得到提高。

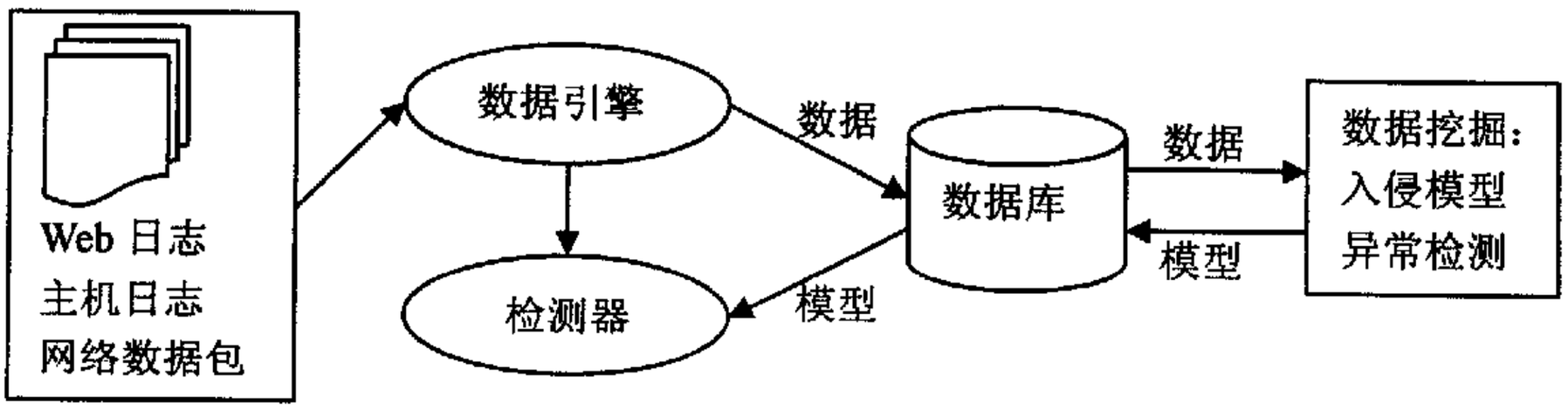

图1-1 基于数据挖掘的入侵检测系统结构

Figurel-1 system structure of DA_DIDS

图 1-1 所示是一个基于数据挖掘的入侵检测系统结构，由数据引擎、检测器、数据库和数据挖掘(生成入侵模型和异常检测等)四部分构成。基于数据挖掘的入侵检测系统首先从原始数据中提取特征，以帮助区分正常数据和攻击行为；然后将这些特征用于模式匹配或异常检测模型；最后提供一种结合模式匹配和异常检测模型的方法。其中，数据引擎观察原始数据并计算用于模型评估的特征；检测器获取数据引擎的数据并利用检测模型来评估它是否是一个攻击；数据库被用作数据和模型的中心存储地；生成入侵模型的主要目的是为了加快开发以及分发新的入侵检测模型的速度。

1.5 课题研究意义和主要研究内容

本文在数据挖掘和分布式入侵检测理论的基础上，提出了一种基于数据挖掘技术的分布式入侵检测模型[7]。主要是在一个分布式入侵检测模型的基础上结合数据挖掘中的聚类分析和关联规则等算法分析网络数据流和主机日志，在比较各类算法实验结果的基础上，针对入侵数据量大、属性值多和实时性要求高等特点改进 k-均值算法和 Apriori 算法[6]，以实现入侵检测系统的数据分析和实时性功能。

在本课题中，我们的研究内容将包括：

1. 分布式入侵检测系统的构成、使用的技术和 CIDF[5]结构框架等。
2. 数据挖掘和分布式思想的引入。
3. 在入侵检测系统中聚类分析和关联规则算法的应用与改进，及其应用于入侵数据分析的实验结果。
4. 整个基于数据挖掘的分布式入侵检测系统的构成，包括系统的组成、软件的设计和实现、实验结果和分析等。

1.6 小结

本章首先分析了课题研究的背景，简要地介绍了目前的数据挖掘技术以及它们在入侵检测系统中的应用及前景。并着重介绍了数据挖掘技术在一个入侵检测系统中的应用，以此提出研究数据挖掘在入侵检测系统中应用的理论与现实意义。本章最后介绍了本课题的主要研究内容和研究意义。

第 2 章　入侵检测系统

入侵检测系统可以弥补防火墙的不足，为网络安全[10]提供实时的入侵检测及采取相应的防护手段，如及时记录证据用于跟踪、切断网络连接，执行用户的安全策略等。入侵检测系统是检测企图破坏计算机资源的完整性、真实性和可用性的行为的软件。入侵检测技术是一种利用入侵者留下的痕迹(如试图登录的失败记录)等信息有效地发现来自外部或内部的非法入侵的技术，它以探测与控制为技术本质，发挥主动防御的作用，是网络安全中极其重要的部分。

2.1 入侵检测系统概述

入侵检测系统的任务就是在提取到的庞大的检测数据中找到入侵的痕迹。入侵分析过程需要将提取到的事件与入侵检测规则进行比较，从而发现入侵行为。一方面入侵检测系统需要尽可能多地提取数据以获得足够的入侵证据，而另一方面由于入侵行为的千变万化而导致判定入侵的规则越来越复杂，为了保证入侵检测的效率和满足实时性的要求，入侵分析必须在系统的性能和检测能力之间进行权衡，合理地设计分析策略，并且可能要牺牲一部分检测能力来保证系统可靠、稳定地运行并具有较快的响应速度。

入侵检测系统在不影响网络性能的情况下能对网络活动进行监测，从而提供对内部攻击、外部攻击和误操作的实时保护。这些主要是通过下任务来实现：

- 监视、分析用户及系统活动；
- 系统构造和弱点的审计；
- 识别反映已知进攻的活动模式并报警；
- 异常行为模式的统计分析；
- 评估重要系统和数据文件的完整性；
- 操作系统的审计跟踪管理，并识别用户违反安全策略的行为。

2.1.1 入侵检测系统分类

入侵检测系统主要是根据入侵行为来分类。入侵(Intrusion)是指任何试图危及资源的安全性、机密性和可用性的行为。这类行为通常分为两类：

1、误用(Misuse)入侵，指利用系统的缺陷进行进攻的行为；

2、异常(Anormaly)入侵，指背离系统正常使用的行为。

相应的入侵检测系统则可以分为三类：

1、采用误用检测的入侵检测系统；

2、采用异常检测的入侵检测系统；

3、采用两种混合检测的入侵检测系统。

一般的入侵检测系统都属于第三类混合型的入侵检测系统。误用入侵通常都有明确的模式，所以误用检测(Misuse Detection)根据对使用系统或网络的信息进行模式匹配来识别使用者的入侵行为。异常检测(Anomaly Detection)则指根据使用者的行为或资源使用状况来判断是否入侵，而不依赖于具体行为是否出现来检测，所以也被称为基于行为的检测。异常检测基于统计方法，使用系统或用

户的活动轮廓(Activity Profile)来检测入侵活动。活动轮廓由一组统计参数组成，通常包括CPU和I/O利用率、文件访问、出错率、网络连接等。这类IDS先产生主体的活动轮廓，系统运行时，异常检测程序产生当前活动轮廓并同原始轮廓比较，同时更新原始轮廓，当发生显著偏离时即认为是入侵。

入侵检测系统还可按照输入数据的来源分为三类：

1、基于主机的入侵检测系统：其输入数据来源于系统的日志，如访问日志、软件使用日志等，一般检测该主机上发生的入侵，主要用于保护关键应用的服务器，实时监视可疑的连接、系统日志检查，非法访问的闯入等。

2、基于网络的入侵检测系统：其输入数据来源于网络各个节点的信息流，能够检测该网段上发生的网络入侵，主要用于实时监控网络关键路径的信息；

3、两部分相结合的分布式入侵检测系统：能够同时分析来自主机系统审计日志和网络数据流的入侵检测系统，一般为分布式结构，由多个部件组成。

完整的入侵检测系统一般分两步：

a)信息收集。

入侵检测的第一步是信息收集，内容包括系统、网络、数据及用户活动的状态和行为。而且，需要在计算机网络系统中的若干不同关键点(不同网段和不同主机)收集信息，这除了尽可能扩大检测范围的因素外，还有一个重要的因素就是从一个源来的信息有可能看不出疑点，但从几个源来的信息的不一致性却是可疑行为或入侵的最好标识。当然，入侵检测很大程度上依赖于收集信息的可靠性和正确性，因此，很有必要只利用所知道的真正的和精确的软件来报告这些信息，因为黑客经常替换软件以搞混和移走这些信息，例如替换程序调用的子程序、库和其它工具。黑客对系统的修改可能使系统功能失常并看起来跟正常的一样，例如；Unix系统的PS指令可以被替换为一个不显示侵入过程的指令，或者是编辑器被替换成一个读取不同于指定文件的程序(黑客隐藏了初始文件并用另一版本代替)。这需要保证用来检测网络系统的软件的完整性，特别是入侵检测系统软件本身应具有相当强的坚固性，防止被篡改而收集到错误的信息。

入侵检测利用的信息一般来自以下四个方面：

1、系统和网络日志文件。黑客经常在系统日志文件中留下他们的踪迹，因此，充分利用系统和网络日志文件信息是检测入侵的必要条件。日志中包含发生在系统和网络上的不寻常和不期望活动的证据，这些证据可以指出有人正在入侵或已成功入侵了系统。通过查看日志文件，能够发现成功的入侵或入侵企图，并很快地启动相应的应急响应程序。日志文件中记录了各种行为类型，每种类型又包含不同的信息，例如记录“用户活动”类型的日志，就包含登录、用户ID改变、用户对文件的访问、授权和认证信息等内容。很显然地，对用户活动来讲，不正常的或不期望的行为就是重复登录失败、登录到不期望的位置以及非授权的企图访问重要文件等等。

2、目录和文件中的不期望的改变。网络环境中的文件系统包含很多软件和数据文件，包含重要信息的文件和私有数据文件经常是黑客修改或破坏的目标。目录和文件中的不期望的改变(包括修改、创建和删除)，特别是那些正常情况下限制访问的，很可能就是一种入侵产生的指示和信号。黑客经常替换、修改和破。坏他们获得访问权的系统上的文件，同时为了隐藏系统中他们的表现及活动痕迹，都会尽力去替换系统程序或修改系统日志文件。

3、程序执行中的不期望行为。网络系统上的程序执行一般包括操作系统、网络服务、用户起动的程序和特定目的的应用，例如数据库服务器。每个在系统

上执行的程序由一到多进程来实现。每个进程执行在具有不同权限的环境中，这种环境控制着进程可访问的系统资源、程序和数据文件等。一个进程的执行行为由它运行时执行的操作来表现，操作执行的方式不同，它利用的系统资源也就不同。操作包括计算、文件传输、设备和其它进程，以及与网络间其它进程的通讯。一个进程出现了不期望的行为可能表明黑客正在入侵你的系统。黑客可能会将程序或服务的运行分解，从而导致它失败，或者是以非用户或管理员意图的方式操作。

4、物理形式的入侵信息。这包括两个方面的内容，一是未授权的对网络硬件连接；二是对物理资源的未授权访问。黑客会想方设法去突破网络的周边防卫，如果他们能够在物理上访问内部网，就能安装他们自己的设备和软件。依此，黑客就可以知道网上的由用户加上去的不安全(未授权)设备，然后利用这些设备访问网络。例如，用户在家里可能安装 Modem 以访问远程办公室，与此同时黑客正在利用自动工具来识别在公共电话线上的 Modem，如果一拨号访问流量经过了这些自动工具，那么这一拨号访问就成为了威胁网络安全的后门。黑客就会利用这个后门来访问内部网，从而越过了内部网络原有的防护措施，然后捕获网络流量，进而攻击其它系统，并偷取敏感的私有信息等等。

b)信号分析

对上述四类收集到的有关系统、网络、数据及用户活动的状态和行为等信息，一般通过三种技术手段进行分析：模式匹配，统计分析和完整性分析。其中前两种方法用于实时的入侵检测，而完整性分析则用于事后分析。

1、模式匹配。模式匹配就是将收集到的信息与已知的网络入侵和系统误用模式数据库进行比较，从而发现违背安全策略的行为。该过程可以很简单(如通过字符串匹配以寻找一个简单的条目或指令)，也可以很复杂(如利用正规的数学表达式来表示安全状态的变化)。一般来讲，一种进攻模式可以用一个过程(如执行一条指令)或一个输出(如获得权限)来表示。该方法的一大优点是只需收集相关的数据集合，显著减少系统负担，且技术已相当成熟。它与病毒防火墙采用的方法一样，检测准确率和效率都相当高。但是，该方法存在的弱点是需要不断的升级以对付不断出现的黑客攻击手法，不能检测到从未出现过的黑客攻击手段。

2、统计分析。统计分析方法首先给系统对象(如用户、文件、目录和设备等)创建一个统计描述，统计正常使用时的一些测量属性(如访问次数、操作失败次数和延时等)。测量属性的平均值将被用来与网络、系统的行为进行比较，任何观察值在正常值范围之外时，就认为有入侵发生。例如，统计分析可能标识一个不正常行为，因为它发现一个在晚八点至早六点不登录的帐户却在凌晨两点试图登录。其优点是可检测到未知的入侵和更为复杂的入侵，缺点是误报、漏报率高，且不适应用户正常行为的突然改变。具体的统计分析方法如基于专家系统的、基于模型推理的和基于神经网络的分析方法，目前正处于研究热点和迅速发展之中。

3、完整性分析。完整性分析主要关注某个文件或对象是否被更改，这经常包括文件和目录的内容及属性，它在发现被更改的、被特洛伊化的应用程序方面特别有效。完整性分析利用强有力的加密机制，称为消息摘要函数(例如 MD5)，它能识别哪怕是微小的变化，其优点是不管模式匹配方法和统计分析方法能否发现入侵，只要是成功的攻击导致了文件或其它对象的任何改变，它都能够发现。缺点是一般以批处理方式实现，不用于实时响应。尽管如此，完整性检测方法还应该是网络安全产品的必要手段之一。例如，可以在每一天的某个特定时间内开

启完整性分析模块，对网络系统进行全面地扫描检查。

2.1.2 入侵检测技术分类

入侵检测技术是以探测与控制为技术本质，发挥主动防御的作用，是网络安全中极其重要的部分，其中的主要智能技术包括：

a）**基于神经网络的入侵检测方法**：这种方法是利用神经网络技术来进行入侵检测。这种方法对用户行为具有学习和自适应功能，能够根据实际检测到的信息有效地加以处理，并做出入侵可能性的判断。该方法还不成熟，目前还没有出现较为完善的产品。

b）**基于专家系统的入侵检测技术**：根据安全专家对可疑行为的分析经验而形成一套推理规则，然后在此基础上建立相应的专家系统，由此专家系统自动对所涉及的入侵行为进行分析。该系统应能随着经验的积累而利用其自学习能力进行规则的扩充和修正。

C）**基于模型推理的入侵检测技术**：根据入侵者在入侵时所执行的某些行为程序的特征，建立一种入侵行为模型，根据这种行为模型所代表的入侵意图的行为特征来判断用户执行的操作是否属于入侵行为。当然这种方法也是建立在对当前已知的入侵行为程序的基础之上的，对未知的入侵方法所执行的行为程序的模型识别需要进一步的学习和扩展。

d）**基于数据挖掘技术的入侵检测技术**：采用的是以数据为中心的观点把入侵检测问题看作为一个数据分析过程。数据挖掘是一种面向应用的技术，同传统的统计概率方法相比，数据挖掘方法具有如下优点：数据挖掘体现了一个完整的数据分析过程，它一般包括数据准备、数据预处理、建立挖掘模型、模型评估和解释等；另外它也是一个迭代的过程，通过不断的调整方法和参数以得到较好的模型。基于数据挖掘的网络入侵检测模型可以进行机器学习和模式扩充，使得手工和经验成分减少，入侵检测效率和可靠性明显得到提高。本文即将利用数据挖掘的算法对分布式入侵检测系统进行性能的改进。

2.1.3 入侵检测系统框架

目前的入侵检测系统大部分是基于各自的需求独立设计和开发的，不同系统之间缺乏工操作性和互用性，这对入侵检测系统的发展造成了障碍，因此DARPA(the Defense Advanced Research Projects Agency，美国国防部高级研究计划局)在 1997 年 3 月开始着手 CIDF(Common Intrusion Detection Framework，公共入侵检测框架)标准的制定。现在加州大学 Davis 分校的安全实验室已经完 CIDF[11]标准，IETF(Internet Engineering Task Force，Internet 工程任务组)成立了 IDWG(Intrusion Detection Working Group，入侵检测工作组)负责建立 IDEF(Intrusion Detection Exchange Format，入侵检测数据交换格式)标准，并提供支持该标准的工具，以更高效率地开发入侵检测系统。国内在这方面的研究刚开始起步，目前也已经开始着手入侵检测标准 IDF(Intrusion Detection Framework，入侵检测框架)的研究与制定。

CIDF 是一套规范，它定义了 IDS 表达检测信息的标准语言以及 IDS 组件之间的通信协议。符合 CIDF 规范的 IDS 可以共享检测信息，相互通信，协同工作，

还可以与其它系统配合实施统一的配置响应和恢复策略。CIDF 的主要作用在于集成各种 IDS 使之协同工作，实现各 IDS 之间的组件重用，所以 CIDF 也是构建分布式 IDS 的基础。

CIDF 的规格文档由四部分组成，分别为：

— 体系结构(The Common Intrusion Detection Framework Architecture)

— 规范语言(A Common Intrusion Specification Language)

— 内部通讯(Communication in the Common Intrusion Detection Framework)

— 程序接口(Common Intrusion Detection Framework APIs)

其中体系结构阐述了一个标准的 IDS 的通用模型；规范语言定义了一个用来描述各种检测信息的标准语言；内部通讯定义了 IDS 组件之间进行通信的标准协议；程序接口提供了一整套标准的应用程序接口(API 函数)。以下将分别对 CIDF 的四个组成部分进行结构分析。

CIDF 的体系结构文档阐述了一个 IDS 的通用模型。它将一个 IDS 分为以下四个组件：

— 事件产生器(Event generators)

— 事件分析器(Event analyzers)

— 事件数据库(Event databases)

— 响应单元(Response Units)

CIDF 将 IDS 需要分析的数据统称为事件(event)，它可以是基于网络的 IDS 从网络中提取的数据包，也可以是基于主机的 IDS 从系统日志等其它途径得到的数据信息。

CIDF 组件之间是以通用入侵检测对象(generalized intrusion detection Objects，以下简称为 GIDO)的形式交换数据的，一个 GIDO 可以表示在一些特定时刻发生的一些特定事件，也可以表示从一系列事件中得出的一些结论，还可以表示执行某个行动的指令。

CIDF 中的事件产生器负责从整个计算环境中获取事件，但它并不处理这些事件，而是将事件转化为 GIDO 标准格式提交给其它组件使用，显然事件产生器是所有 IDS 所需要的，同时也是可以重用的。CIDF 中的事件分析器接收 GIDO，分析它们，然后以一个新的 GIDO 形式返回分析结果。CIDF 中的事件数据库负责 GIDO 的存贮，它可以是复杂的数据库，也可以是简单的文本文件。CIDF 中的响应单元根据 GIDO 做出反应，它可以是终止进程、切断连接、改变文件属性，也可以只是简单的报警。

CIDF 将各组件之间的通信划分为三个层次结构：GIDO 层(GIDO Layer)。消息层(Message Layer)和传输层(Negotiated Transport Layer)。其中传输层不属于 CIDF 规范，它可以采用很多种现有的传输机制来实现。消息层负责对传输的信息进行加密认证，然后将其可靠地从源传输到目的地，消息层不关心传输的内容，它只负责建立一个可靠的传输通道。GIDO 层负责对传输信息的格式化，正是因为有了 GIDO 这种统一的信息表达格式，才使得各个 IDS 之间的互操作成为可能。

CIDF 也对各组件之间的信息传递格式、通信方法和标准 API 进行了标准化。在现有的 IDS 中，经常用数据采集部分、分析部分、响应部分和日志来分别代替事件产生器、事件分析器、响应单元和事件数据库这些术语。

CIDF 的规范语言文档定义了一个公共入侵标准语言(Common Intrusion

Specification Language，以下简称为 CISL)，各 IDS 使用统一的 CISL 来表示原始事件信息、分析结果和响应指令，从而建立了 IDS 之间信息共享的基础。CISL 是 CIDF 的最核心也是最重要的内容。

要建立一个 CISL 并非易事，它必须能够满足以下的要求：

- 有丰富的表达能力，能够表示各种入侵行为及其相应的处理方法；
- 表达唯一，表达本身不能产生二义性；
- 语言精确，对每一种表达的含义做出明确的定义；
- 可分层次，不同层次的组件能够得到不同的信息；
- 可自定义，能够解释自己需要得到的信息；
- 高效率，尽可能少地站用系统资源；
- 可扩展，可以增加新的表达内容而不影响整个语言结构；
- 语言简朴，语言可以被快速编码和解码；
- 可移植，语言能够支持多种操作系统平台及多种传输机制；
- 易于实现，如果语言复杂到无法实现，语言本身也就失去意义了。

为了能够满足以上的要求，CISL 使用了一种类似 Lisp 语言的 S 表达式，它的基本单位由语义标志符(Semantic IDentifiers，以下简称为 SID)、数据和圆括号组成。例如。(HostName “first.example.com”)，其中 HostName 为 SID，表示后面的数据是一个主机名，“first.example.com”伪数据，括号将两者关联。

多个 S 表达式的基本单位递归组合在一起，构成整个 CISL 的 S 表达式。在 CISL 中，所有信息(原始事件、分析结果、响应指令等)均是用这种 S 表达式来表示的。例如下面的 S 表达式

```
(Delete
    (Context
     (HostName' first. example. com' )
     (Time' 16: 40: 32 Jun 14 1998' )
  )
(Initiator
     (Username'  joe' )
  )
     (Source
     (FileName' / etc / passwd' )
       )
  )
```

表示用户名为“joe”的用户在 1998 年 6 月 14 日 16 点 40 分 32 秒删除了主机名为“first.example.com”的主机上的文件 “etc / passwd”。这是一个事件描述，其中的 Delete、Context、HostName、Time 等均为 SID。

在 CIDF 的规范语言文档附录 A 中有关于 SID 的详尽描述，其中名字为“AttackID”的 SID 表示了一个分类的入侵类型列表。

这些 SID、S 表达式以及 S 表达式的组合。递归、嵌套等构成了 CISL 的全部表达能力，也即 CISL 本身。在计算机内部处理 CISL 时，为节省存储空间提高运行效率，必须对 ASCII 形式的 S 表达式进行编码，将其转换为二进制字节流的形式，编码后的 S 表达式就是 GIDO。GIDO 是 S 表达式的 M 进制形式，是 CIDF 各组件统一的信息表达格式，也是组件之间信息数据交换的统一形式。在 CISL 中还对 GIDO 的动态追加，多个 GIDO 的组合，以及 GIDO 的头结构等进行了定义等。

目前CIDF还没有成为正式的标准，也没有一个商业IDS产品完全遵循该规范，但各种IDS的结构模型具有很大的相似性，各厂商都在按照CIDF进行信息交换的标准化工作，有些产品已经可以部分地支持CIDF。可以预测，随着分布式IDS的发展，各种IDS互操作和协同工作的迫切需要，各种IDS必须遵循统一的框架结构，CIDF将成为事实上的IDS的工业标准。

2.2 传统入侵检测方法

目前，国际顶尖的入侵检测系统IDS主要以模式发现技术[13]为主。所谓模式发现技术是指从各种入侵行为及其变种中抽取出各方面的模式或特征，然后用这些模式去匹配、去发现可能的攻击。IDS一般从实现方式上分为两种：基于主机的IDS和基于网络的IDS。一个完备的入侵检测系统IDS应该是基于主机和基于网络两种方式兼备的分布式系统。另外，能够识别的入侵手段的数量多少，最新入侵防范的更新是否及时都是评价入侵检测系统的关键指标。

传统的入侵检测有基于主机、基于网络、基于统计模型等几类，但通常都按照基于主机的IDS和基于网路的IDS来划分。基于主机的IDS原理上类似于计算机防病毒软件或是网络管理软件——在所有服务器上安装某种代理，用特定的管理控制系统进行汇报工作。它们使用一种传感控制结构，而且通常是纯被动的。基于网络的系统可以通过比较线路的流量，以及内部列出的种种特征标识进行观测。两种方案都可以对识别到的攻击做出反应。

2.2.1 基于主机的入侵检测

就目前的情况来看，DNS、Email和web服务器是多数网络攻击的目标，大约占据全部网络攻击事件的1/3以上。这些服务器必须要与Internet系统交互作用，所以应当在各服务器上安装基于主机的入侵检测软件，其检测结果也应及时地向管理员报告。基于主机的IDS没有带宽的限制，它们密切监视系统日志，能识别运行代理的机器上受到的攻击。基于主机系统提供了基于网络系统不能提供的精细功能，包括二进制完整性检查、系统日志分析和非法进程关闭等功能。

基于主机的IDS最大的好处就在于能根据受保护站点的实际情况进行针对性的定制，使其工作非常有效果，误警率相当低。典型的如web服务器入侵检测系统，它相当于一套复杂的过滤设备，使用一个攻击字符串列表来对web服务器(单个或多个)进行监视，可以发现对web服务器的已知的各种可能的攻击。这样的IDS在工作时，即使有一些误警事件也关系不大，因为这样可以通告管理员在web服务器上运行了哪些脆弱的服务，从而采取打补丁或升级应用程序的对策。管理员甚至还可以自己编写或升级这样的IDS。

在现实的网络世界里，利用操作系统的漏洞或应用软件的缺陷进行网络攻击的事件几乎每天都在发生，甚至已经成为一些技术并不高明但又喜欢恶作剧的上网用户的娱乐节目。人们也很容易从专门的黑客网站上获得关于Windows、Linux、Netware等各种网络操作系统的漏洞报告，或是收集到各类DNS、Email、web服务等应用软件的各种缺陷，并且很快搜集到各种各样的黑客软件进行“实践”。基于主机的入侵检测系统已经在此方面起到了越来越重要的防范效果。

2.2.2 基于网络的入侵检测

早期的 IDS 模型设计用来监控单一服务器，是基于主机的入侵检测系统；然而近来更多的模型集中用于监控通过网络互连的多个服务器和客户机，是基于网络的入侵检测系统。

基于网络的系统具有强制性而且是独立于平台的，部署它们对网络影响很小或没有影响。只对带宽有较高要求。基于网络的 IDS 依靠对网络数据包和网络故障等的分析来进行检测，往往能对异常情况做出较快的响应。基于网络的入侵检测通常采用多个 IDS 检测器配合一个 IDS 指示器的分布式结构来协同工作。在网络 IDS 方面，Internet Security System 公司的 IDS 产品 RealSecure 在市场上占有主导地位。Cisco 的 NetRanger，北京中洲基业软件技术有限公司的网警 NetCop，中科院高能所网络工作室的天眼，以及启明星辰公司的 SkyBell 和天阗等亦属于基于网络 IDS 产品，并且这些网络 IDS 在结构上都大体相同。

与基于主机的 IDS 高命中率不同，现有的基于网络的 IDS 在所有的检测结果中，误警率都高达甚至有 90%以上，这类 IDS 需要经过专门培训的网络分析员才能真正发挥效果。IDS 检测器(又叫产生器或引擎)分布在网络上受保护的各个服务器和客户机上，并且这些服务器和客户机可能采用的是各式各样的操作系统。IDS 检测器根据定义好的策略进行事件检测，并将检测结果汇总到 IDS 指示器综合。IDS 指示器还负责以特有的协议向各 IDS 检测器发布策略。面对海量的网络数据，IDS 系统经常要求配备高性能的机器，从检测器到指示器，都需要大容量的内存、高速的中央处理器和大容量快速硬盘，并且这些计算机有可能几个月就需要更新，交给办公人员作普通的客户机使用。

2.3 传统入侵检测方法的缺陷

基于主机的入侵检测系统可以精确地判断入侵事件，并可对入侵事件立即进行反应，还可针对不同操作系统的特点判断应用层的入侵事件，其缺点是与操作系统和应用层软件结合过于紧密，通用性可能会较差，并且 IDS 的分析过程会占用主机宝贵的资源。

基于网络的入侵检测系统只能监视经过本网段的活动，并且精确度较差，在交换式网络环境下难于配置，防入侵欺骗的能力也比较差。严格地说，如果要使网络得到高水平的安全保护，就应该选择更加主动和智能的网络安全技术。完备的网络安全系统应该能够监视网络和识别攻击信号，能够做到及时反应和保护，能够在受到攻击的任何阶段帮助网络管理人员从容应付。

就目前的研究和开发经验来看，很难建立一个模型来判定一个特定的网络连接是善意的还是恶意的；即使是目前国际上最先进的入侵检测系统，对某些攻击类型的误警率也高达 90%以上。所以，现有的入侵检测，无论是基于主机的 IDS 还是基于网络的 IDS，在很大程度上都依赖于网络管理员的能力和直觉，需要人工地调整入侵检测系统。攻击事例和管理员的经验是入侵检测中最为重要的两个方面，IDS 的开发与升级需要攻击事例和管理员经验这两方面长时间的原始积累。

大多数入侵检测系统只能寻找它们已知的攻击类型。但很多安全产品供应商已经开始让他们的入侵检测产品理会系统遇到的事先没有定义属性的异常。当然，入侵检测系统和攻击评估工具有共同的限制，它们都可以被新的攻击手段突

破。因此，对它们不断地升级更新是至关重要的。

在 IDS 的升级方面，各种现有的知名产品大体上相同。美国能源部开发的官方 IDS 软件 NID 允许用户自己编写升级的特征。ISS 公司的 RealSecure 不允许用户自己创建新的过滤规则，而是自己每月发布一次供升级用的攻击特征。Cisco 公司的 Network Fighter Recoder 既允许用户自己创建过滤器，用户又可以在销售商或者在网站上获得新的升级软件。信息产业部电子第十五研究所开发的 reep 系统亦是依靠代码特征库升级的办法来检测新出现的攻击类型。

由上可见，对事例和经验的依赖也反映为现有的 IDS 都缺乏自学习、自适应能力。即使是那些能响应事先未定义属性的异常事件的智能化 IDS，也普遍存在漏警率、误警率都高居不下的困难。

此外，99 年后半年以来，分布式拒绝服务攻击(Distributed Denial of Service Attack)也开始成为网络黑客们越来越青睐的攻击手段。进入 2000 年以来，网络遭受这类 DDOS 攻击的事件不断发生，全球许多著名网站如 yahoo、cnn、buy、ebay、fbi，包括中国的新浪网都相继遭到不名身份的黑客攻击。在这些攻击行为中，黑客摈弃了以往常常采用的更改主页这一对网站实际破坏性有限的做法，取而代之的就是这种在一定时间内使被攻击的网络彻底丧失正常服务功能的 DDOS 手法；网络攻击不再像以往来自单一的主机，而是来自不同子网的数目巨大的主机群。目前国内外还没有成熟产品能彻底防御这种攻击，甚至很难找出有效的防范方案。由此也反映出，入侵活动可以具有很大的时间跨度和空间跨度，入侵的手段也越来越复杂和先进。Internet 的发展，永远伴随着无休止的网络安全攻与防的较量。

2.4 入侵检测技术的发展方向

在入侵检测技术发展的同时，入侵技术也在更新，一些地下组织已经将如何绕过 IDS 或攻击 IDS 系统作为研究重点。高速网络，尤其是交换技术的发展以及通过加密信道的数据通信，使得通过共享网段侦听的网络数据采集方法显得不足，而大量的通信量对数据分析也提出了新的要求。近年对入侵检测技术有几个主要发展方向：

(1)分布式入侵检测与通用入侵检测架构

传统的 IDS 一般局限于单一的主机或网络架构，对异构系统及大规模的网络的监测明显不足。同时不同的 IDS 系统之间不能协同工作能力，为解决这一问题，需要分布式入侵检测技术与通用入侵检测架构。CIDF 以构建通用的 IDS 体系结构与通信系统为目标，GrIDS(Graph-based Intrusion Detection System)的设计目标是大规模自动或协同攻击的入侵检测。

(2)应用层入侵检测

许多入侵的语义只有在应用层才能理解，而目前的 IDS 仅能检测如 Web 之类的通用协议，而不能处理如 LotusNotes、数据库系统等其他的应用系统。许多基于客户、服务器结构与中间件技术及对象技术的大型应用，需要应用层的入侵检测保护。Stillerman 等人已经开始对 CORBA 的 IDS 研究。

(3)智能的入侵检测

入侵方法越来越多样化与综合化，尽管已经有智能体、数据挖掘在入侵检测领域应用研究，但是这只是一些尝试性的研究工作，需要对智能化的 IDS 加以进一步的研究以解决其自学习与自适应能力。

(4)入侵检测的评测方法

用户需对众多的 IDS 系统进行评价，评价指标包括 IDS 检测范围、系统资源占用、IDS 系统自身的可靠性与鲁棒性(所谓“鲁棒性”，是指系统在一定(结构，大小)的参数摄动下，维持某些性能的特性。也即“抗干扰能力”)。从而设计通用的入侵检测测试与评估方法与平台，实现对多种 IDS 系统的检测已成为当前 IDS 的另一重要研究与发展领域。

2.5 小结

本章首先简单了 IDS 的概念以及标准，对 IDS 的类型也做了一个简单分类。然后详细地介绍了 IDS 的基本结构，所面临的问题以及发展趋势，我们的基于数据挖掘的分布式入侵检测就是在该基础之上推出的。总之，为给网络应用提供可靠服务，对网络通信技术安全性的要求越来越高，而由于入侵检测系统能够从网络安全的立体纵深、多层次防御的角度出发提供安全服务，必将进一步受到人们的高度重视。

第 3 章　基于数据挖掘的分布式入侵检测思想的引入

3.1 分布式思想

3.1.1 分布式入侵

一个典型的分布式入侵行为的可以用如下图 3-1 来表示。

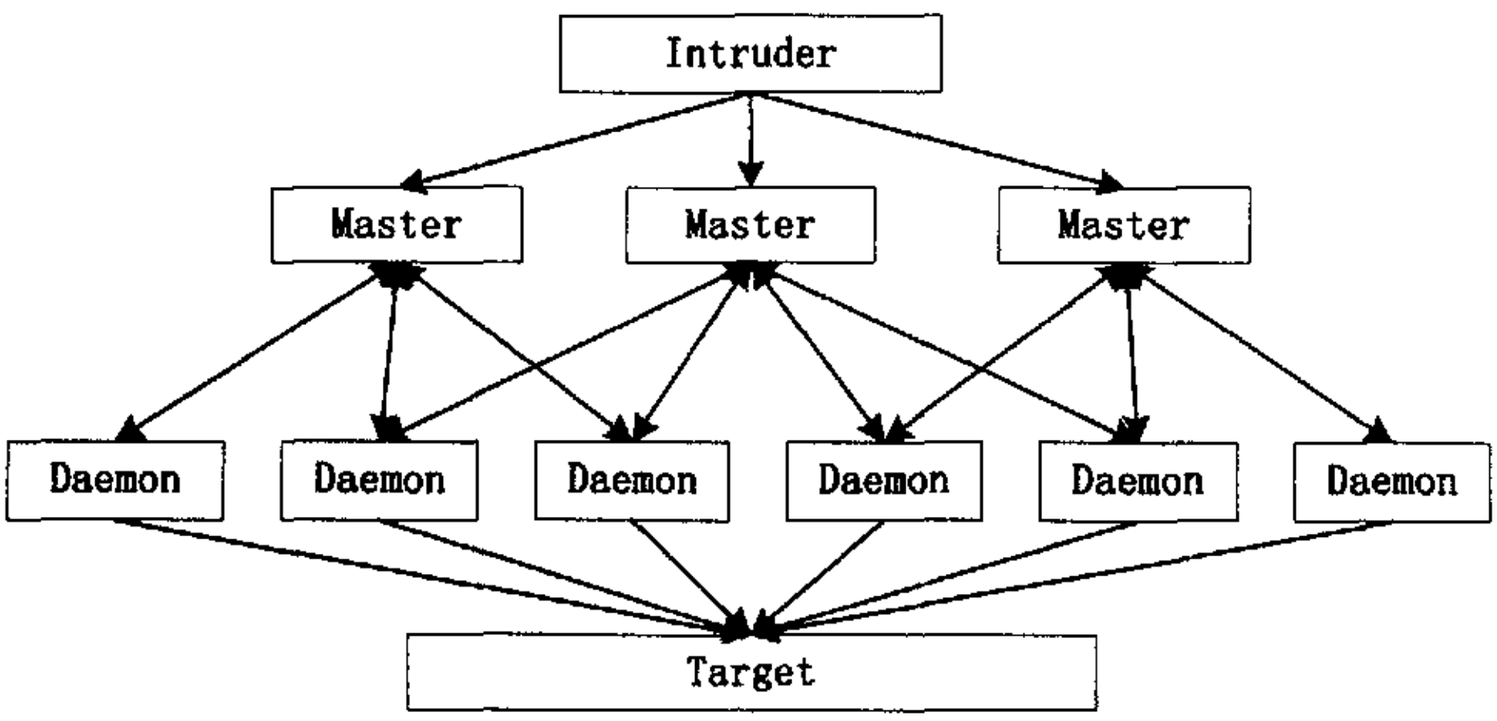

图3-1 分布式入侵模型

Figure 3-1 A Model of Distributed Intrusion

入侵者(Intruder)通过少数的几台主控机器(Master)入侵到大量的主机(Daemon)内部，这些主机可以广泛地分布在较大范围的网络中，然后用 Master 控制被入侵主机 Daemon，向目的主机(Target)发起分布式的攻击。在 Target 看来，从各个 Daemon 来的请求几乎都是合法的。但是由于 Daemon 的数量可以很大(几百甚至几千)，这个机群同时向 Target 发起攻击的时候，可以对 Target 造成致命的打击，这就是分布式入侵的原理。

任何基于单一主机的防护措施，在这样的攻击面前都显得如此苍白无力。由于攻击性质已经发生了改变，整个网络成为攻击目标，单一主机的安全性在整体网络的脆弱性面前显得不堪一击，很多 IDC 机房全面瘫痪，一些安全设置极高的 Unix 服务器也无法在这样的网络环境里幸免，这在以前是非常罕见的。

3.1.2 分布式入侵检测

通过对传统入侵检测模型的分析，我们可以看到，各种检测方法模型和技术手段都有其局限性，没有比较通用的检测方法一个解决方法就是对不同的检测环境进行分类，对不同的环境采用不同的检测方法和技术手段这就是分布式入侵检测的思想，它采用多个检测部件，各检测部件选用不同的检测方法，协同合作，完成检测任务这有利于取各种检测方法之长，以大幅度地提高检测效率和准确性。

到目前为止，已有一些研究机构对分布式入侵检测进行了有益的研究，也建

立了一些实验性系统 UCDavis 的 GrIDS[14]在每台主机上运行一个模块，它们向一台固定的机器发送信息，然后在这台机器上建立网络活动的图形描述，用这个描述来检测可能的入侵。

其分布式入侵检测模型如图 3-2 所示。

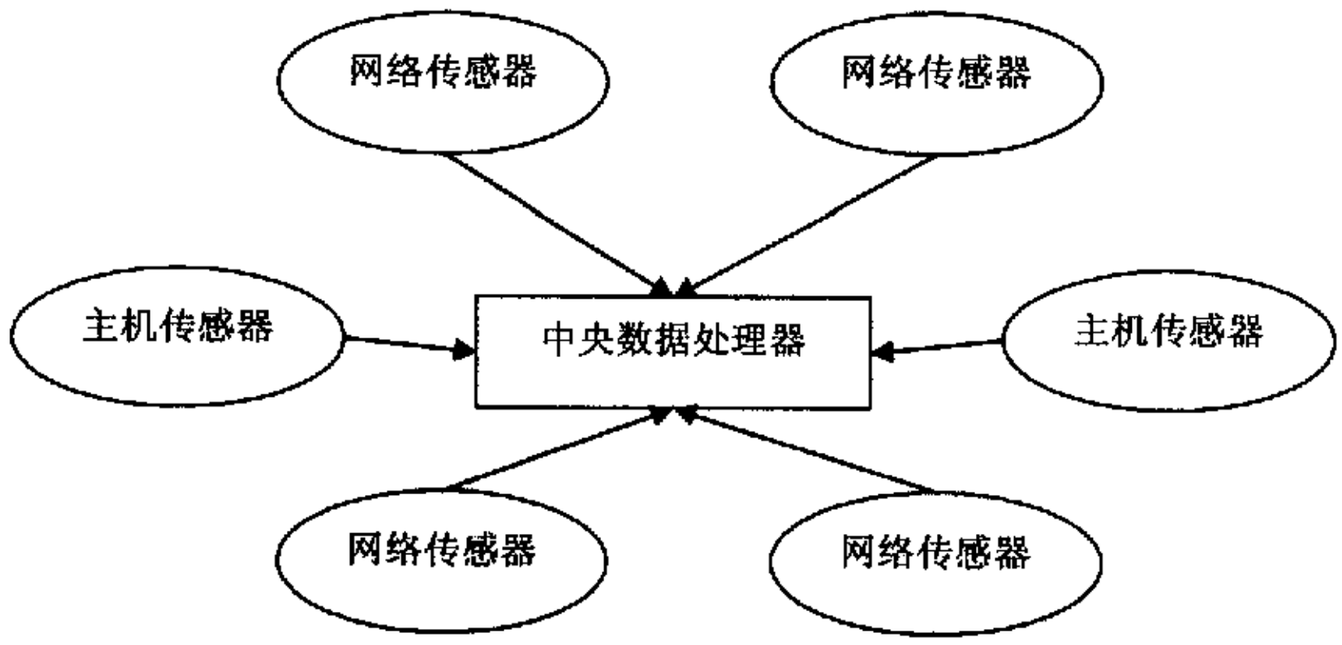

图3-2 UCDavis的分布式入侵检测系统GrIDS
Figure 3-2 GrIDS of UCDavis

该种体系结构将以往基于主机和基于网络的入侵检测系统结合起来。整个系统包括三个部分，位于每台监控主机上的传感器，局域网上的网络传感器和中央数据处理器。主机传感器和网络传感器分别从主机和局域网上采集有用数据，然后将数据送至中央数据处理器作全局的入侵检测。这个体系结构侧重于网络用户识别问题的解决，也就是通过跟踪用户在网络上的移动情况，计算用户操作的相关性来判断是否有入侵行为发生。利用分布式部件进行数据收集，收集到的数据最后被传送到一个处理中心，在处理中心进行统一处理。这种结构使检测系统无法达到实时性。而且，由于检测部件依赖于数据收集部件，如果数据收集部件或检测部件出错，都会影响系统的正常运行。这个体系结构运用范围较小，而且由于模型过于简单，无法检测复杂的入侵行为。要使系统具有更强的检测能力，需要进一步完善系统体系结构和检测模型。

3.2 数据挖掘思想

3.2.1 数据挖掘的任务

信息技术和因特网的迅速发展使得信息的增长量相当巨大，入侵方法越来越多样化与综合化，分布式思想虽然可以提高检测的效率，但是面对不断发展的黑客攻击技术，把庞大的入侵规则的特征提取仅交给专家去做就显得力不从心，这就要求一种新的技术来管理这些信息，并从中及时发现有用的知识，提高信息的利用率。数据挖掘(Data Mining)是从大量的、不完全的、有噪声的、模糊的、随机的数据中，提取隐含在其中的人们事先不知道的、但又是潜在有用的信息和知识的过程。

与数据挖掘相近的同义词有数据融合、数据分析和决策支持等。这个定义包括好几层含义：数据源必须是真实的、大量的、含噪声的；发现的是用户感兴趣的知识；发现的知识要可接受、可理解、可运用；并不要求发现放之四海皆准的知识，仅支持特定的发现问题。

何为知识？从广义上理解[15]，数据、信息也是知识的表现形式，但是人们更把概念、规则、模式、规律和约束等看作知识。人们把数据看作是形成知识的源泉，好像从矿石中采矿或在矿砂中淘金一样。原始数据可以是结构化的，如关系数据库中的数据；也可以是半结构化的，如文本、图形和图像数据；甚至是分布在网络上的异构型数据。

数据挖掘的任务是从数据中发现模式[16]。模式有很多种，按功能可分为两大类：预测型(Predictive)模式和描述型(Descriptive)模式。预测型模式是可以根据数据项的值精确确定某种结果的模式。挖掘预测型模式所使用的数据也都是可以明确知道结果的。例如，根据各种动物的资料，可以建立这样的模式：凡是胎生的动物都是哺乳类动物。当有新的动物资料时，就可以根据这个模式判别此动物是否是哺乳动物。描述型模式是对数据中存在的规则做一种描述，或者根据数据的相似性把数据分组。描述型模式不能直接用于预测。例如，在地球上，70%的表面被水覆盖 30%是土地。

在实际应用中，往往根据模式的实际作用细分为以下 6 种：

1) 分类模式

2) 回归模式

3) 时间序列模式

4) 聚类模式

5) 关联模式

6) 序列模式

在我们的入侵检测系统中，我们主要应用了聚类模式和关联模式。

聚类模式[17]把数据划分到不同的组中，组之间的差别尽可能大，组内的差别尽可能小。与分类模式不同，进行聚类前并不知道将要划分成几个组和什么样的组，也不知道根据哪一(几)个数据项来定义组。一般来说，业务知识丰富的人应该可以理解这些组的含义，如果产生的模式无法理解或不可用，则该模式可能是无意义的，需要回到上阶段重新组织数据。

关联模式[18]反映一个事件和其他事件之间依赖或关联的知识。如果两项或多项属性之间存在关联，那么其中一项的属性值就可以依据其他属性值进行预测。最为著名的关联规则发现方法是 R. Agrawal 提出的 Apriori 算法。关联规则的发现可分为两步。第一步是迭代识别所有的频繁项目集，要求频繁项目集的支持率不低于用户设定的最低值；第二步是从频繁项目集中构造可信度不低于用户设定的最低值的规则。识别或发现所有频繁项目集是关联规则发现算法的核心，也是计算量最大的部分。

挖掘关联规则的目标是从数据表中找出多种特征属性之间的联系。它的一个简单而且有趣的商业应用就是根据购物的记录来发现顾客什么商品会一起买，然后据此来安排货品的摆放。通常来说，给定了一系列的记录，每个记录都是很多项的组合，关联规则给出的表达形式就是 X→Y，支持度和置信度。其中 X 和 Y 是记录项的子集，支持度是记录中包含 X+Y 的百分比，而置信度是(X+Y)的置信度与 X 的置信度的比值。

例如，有关联规则 news→politics[0.3，0.1]，它源自于一个网站的 log 文件。它的意思是在读者在打开 news 时，有 30%的情况也同时打开了 politics，而在所有的数据中打开 news 的情况占 10%。

应用关联规则分析到入侵检测系统的目的以及一般过程是：

(1)特征抽取与数据预处理。审计数据和网络数据能够被整理到数据库的表

中，其中每一行都是一条数据记录，而每一列都是一种系统特征(这个特征有可能是经过预处理的统计特征)；

(2)关联规则挖掘分析。很多事实表明程序的执行与用户的行为之间存在着频繁的联系。例如很多用户在进行越权操作时，一般是程序对特定的目录或文件进行更改，这种情况发生的很多；

(3)检测入侵。将新产生的关联规则添加到关联规则库中去，然后将用户行为与关联规则库中的规则匹配来判断是否入侵。

3.2.2 数据挖掘的步骤和流程

数据挖掘的处理对象是大量的数据，这些数据一般存储在数据库系统中，是长期积累的结果，但往往不适合直接在这些数据上面进行知识挖掘，需要做数据准备工作，一般包括数据的选择(选择相关的数据)、净化(消除噪音、冗余数据)、推测(推算缺失数据)、转换(离散值数据与连续值数据之间的相互转换，数据值的分组分类，数据项之间的计算组合等)、数据缩减(减少数据量)。如果数据挖掘的对象是数据仓库，那么这些工作往往在生成数据仓库时已经准备妥当。数据准备是数据挖掘的第一个步骤，也是比较重要的一个步骤。数据准备是否做好将影响到数据挖掘的效率和准确度以及最终模式的有效性。

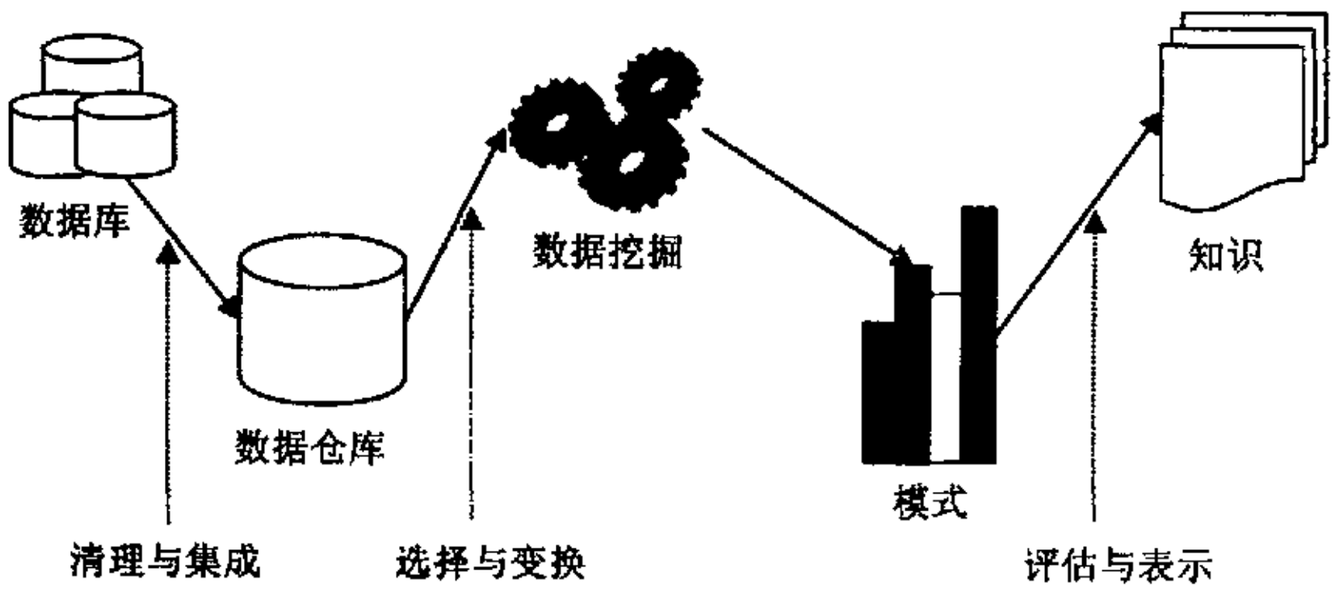

图3-3 数据挖掘作为知识发现的步骤
Figure 3-3 steps of KDD

通常把数据挖掘认为是数据库知识发现的一个基本步骤。知识发现过程[19]如图 3-3 所示，由以下步骤组成：

1)数据清理(消除噪声和不一致数据)

2)数据集成(多种数据源可以组合在一起)

3)数据选择(从数据库中检索和分析任务相关的数据)

搜索所有与业务对象有关的内部和外部数据信息，并从中选择出适用于数据挖掘应用的数据，研究数据的质量，为进一步的分析做准备，并确定将要进行的挖掘操作的类型。

4)数据变换(数据交换或统一成适合挖掘的形式，如通过汇总和聚操作)

将数据转换成一个分析模型，这个分析模型是针对挖掘算法建立的，建立一个真正适合挖掘算法的分析模型是数据挖掘成功的关键。

5)数据挖掘(基本步骤，使用智能方法提取数据模式)

对所得到的经过转换的数据进行挖掘，选取相应算法的参数，分析数据，得到可能形成知识的模式模型。除了完善选择合适的挖掘算法外，其余一切工作都能一动地完成。

6) 模式评估(根据某种兴趣度度量，识别表示知识的真正有趣的模式)
解释并评估结果，其使用的分析方法一般应作数据挖掘操作而定，通常会用到可视化技术。用户理解的、并被认为是符合实际和有价值的模式模型形成了知识。同时还要注意对知识做一致性检查，解决与以前得到的知识互相冲突、矛盾的地方，使知识得到巩固。

7) 知识表示(使用可视化和知识表现技术，向用户提供挖掘的知识)
将分析所得到的知识集成到业务信息系统的组织结构中去。运用知识有两种方法；一种是只需看知识本身所描述的关系或结果，就可以对决策提供支持；另一种是要求对新的数据运用知识，由此可能产生新的问题，而需要对知识做进一步的优化。数据挖掘过程可能需要多次的循环反复，每一个步骤一旦与预期目标不符，都要回到前面的步骤，重新调整，重新执行。

数据挖掘是一个完整的过程，该过程从大型数据库中挖掘先前未知的、有效的、可实用的信息，并使用这些信息做出决策或丰富知识。

3.2.3 在入侵检测中引入数据挖掘技术

将数据挖掘应用在入侵检测系统中能够有效的发现用户行为模式，对于入侵的检测能力很强，而且对某些从未出现过的入侵方式也能有很好的检测效果。文章提出了一种基于数据挖掘的系统结构模型，将数据挖掘与分布式的入侵检测方法结合[20][21]起来，并引入了 Agent 的概念[22]，有很大的灵活性与可扩展性。

在入侵检测中使用数据挖掘方法的最大挑战在于为了计算出准确的规则集，需要大量的审计数据。传统的方法[23]是为目标系统的每一种资源计算检测模型，这就加重了数据挖掘的任务.被检测模型所有的规则集并不是静止的，因此挖掘的过程也是一个不断学习的过程.一方面，数据挖掘能够有效发现入侵行为，但是它本身非常消耗时间和存储量，而另一方面一个成功的 IDS 要求对不断出现的入侵做出实时的响应.如何在有效性和实时性之间找到平衡点，是值得我们探究的问题.

IDS 不仅需要高的精确度，还需要好的可扩展性和适应性。一段网络可能会有很多个入侵点，例如恶意的 IP 包可能在瞬间摧毁一个目标主机，或者是系统软件的一个漏洞可能导致根权限的丧失。既然这些攻击点可能已经在某些地方被发现过，就需要把这些信息随时的扩散出去。所以 IDS 的及时更新也是十分重要的。

目前构造一个 IDS 是一项十分宏大的工程。很多情况下专业人员需要对已知的攻击方法和已知的系统漏洞进行分析，提取特征，然后用到误用入侵检测中去。或者是采用很多种统计分析方法来做异常检测，但是可扩展性和适应性都不强。而且很多的 IDS 只能够处理某些特定类的数据源，而且更新十分缓慢和昂贵。鉴于以上原因，将数据挖掘引入入侵检测系统中，它的自动发现数据特征的功能在入侵检测中也取得了很好的效果。

入侵检测的目标是建立一个功能完备的扩展性好的系统，要求能够便于配置、更新和扩展，不依赖于平台。该文提出的基于数据挖掘的分布式 IDS 模型[24]采用 Agent 的方式:学习 Agent 和检测 Agent 之间通过规则库相连。学习 Agent 各自采用不同数据挖掘的方法不断的学习，并将学习的结果存入规则库，检测 Agent 通过误用检测，高效率的检测入侵行为，最后由检测 Agent 判断是否入侵。

对于一个网段来说，每种 Agent 在内部机制上又分为基于主机的和基于网络的两种。各个Agent 相互之间通过通信Agent 传递需要的数据和信号。除了学习 Agent 对规则库不断的更新之外，还可以由网络来学习。学习 Agent 对整个网段的数据进行分析学习，发现对整个网段的入侵，然后由通信 Agent 将信息反馈到各个检测 Agent，再次判断是否入侵。

这样的一个体系结构的主要优势在于：

(1)容易建立一个有层次的 IDS。

(2)检测代理可以更加轻松地工作。因为大量的学习和计算工作都交给了学习代理去做，而检测代理依据学习代理输入的信息独立地工作，这就使其能更加有效地专注于检测。

(3)一个检测代理通过传输新的审计记录到学习代理可以报告新的入侵事例，而学习代理又反过来升级分类器去检测新的入侵，把这些入侵信息及时地发送到所有的基础检测代理。

3.3 小结

本章我们首先看到了一个分布式的入侵行为及其破坏性，进而我们引入了基于 Agent 的分布式入侵检测系统框架，然而现在的入侵技术日新月异，为了及时的更新我们的入侵规则库，本文又引入了数据挖掘思想，通过对本机和网络数据的分析来发现新的入侵规则，使入侵检测系统具有了自学习的能力。

第 4 章　DA_DIDS 系统的结构设计

4.1 DA_DIDS 系统结构模型

4.1.1 DA_DIDS 整体模型设计

图 4-1 为基于数据挖掘的分布式入侵检测系统结构示意图。左部为内部网的一般节点，右部为 IDS 服务器。最底层的是多个数据采集 Agent[33]。左部的一般节点用多个数据采集 Agent 收集本地系统数据和进出该节点的数据，然后由本地数据挖掘 Agent 进行分析。分析得到的规则存入本地规则库。同时，进出节点的数据还通过通信 Agent 传送到 IDS 服务器，以便进行综合的分析。为了提高入侵检测的速度，收集到的数据同时送入检测 Agent 进行实时检测，检测的规则来自于本地的规则库。而数据挖掘 Agent 用于对数据进行离线分析，再将分析的结果存入规则库。本地规则库同时也能收到 IDS 服务器的的一些更新信息。检测结果对数据挖掘 Agent 存在一个反馈，它们之间的通信通过通信 Agent 来实现。这种情况是对于性能较好的节点，因为离线的分析会耗用大量的资源。如果节点性能较差，就可能取消数据挖掘 Agent，而本地规则库直接来源于 IDS 服务器。

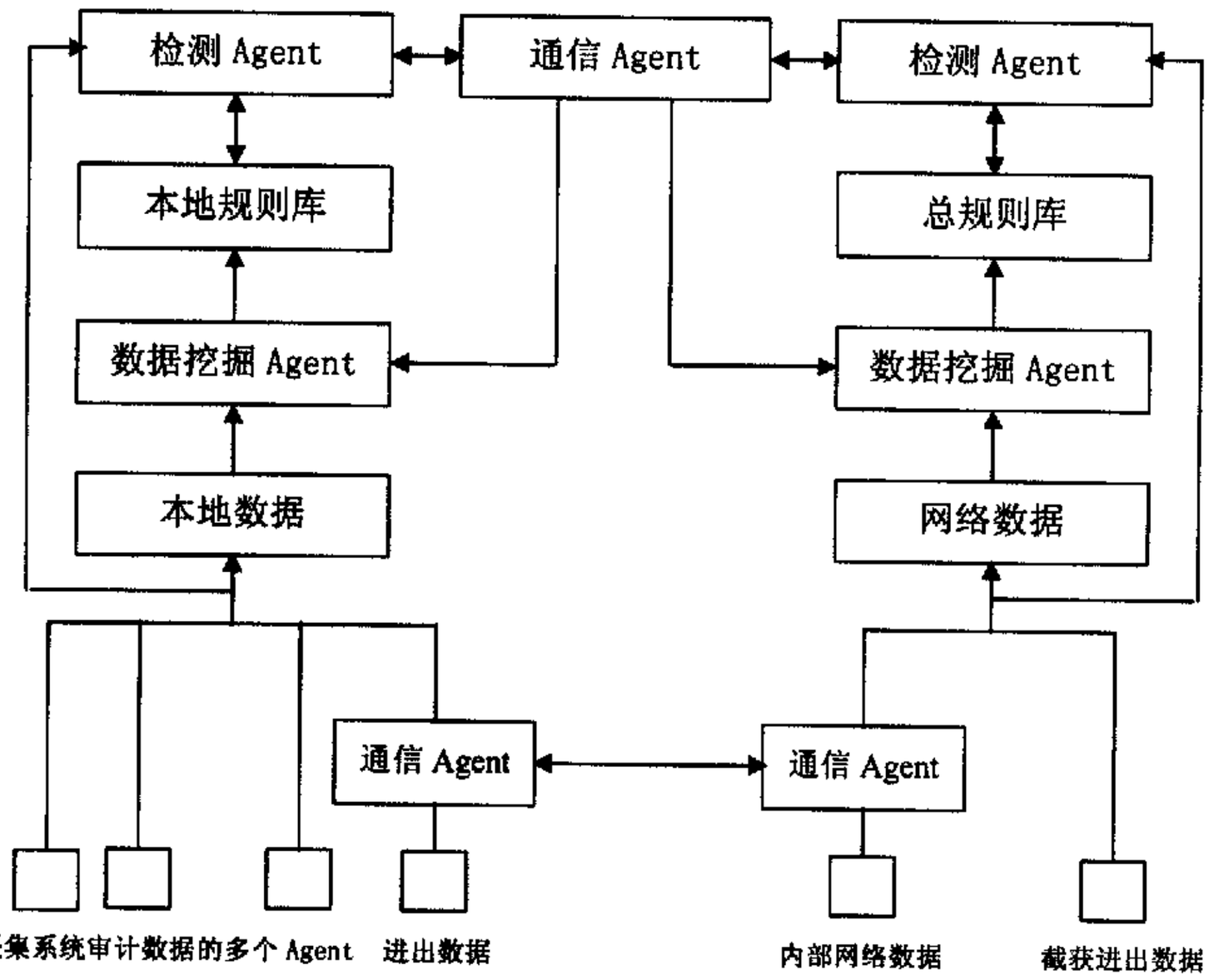

图4-1　基于数据挖掘的分布式入侵检测系统结构示意图

Figure 4-1 Agent structure of DA_DIDS

右部的 IDS 服务器基本体系结构与节点比较类似，不过它的数据来源有两

类，一个是网关处收集的网络数据，另一部分是各个节点发来的相关内部数据。进行分析后，存入总规则库，并随时对分规则库保持通信。IDS 服务器和节点检测 Agent 也存在一个协调裁决的问题。

这种体系结构[35]的优点是，将较大负载的网络数据集中到 IDS 服务器处理，而本地数据则就近处理，避免了内部网中繁重的数据交流。同时还分析了网内数据，可有效防范一些内部较复杂的入侵(如跳板)等问题。

4.1.2 DA_DIDS 的各模块设计

4.1.2.1 DA_DIDS 的代理结构体系

我们所提出的基于数据挖掘的分布式入侵检测模型[44]采取多 Agent 结构，每个检测部件都是独立的检测单元，尽量降低各检测部件间的相关性，不仅实现了数据收集的分布化，而且将入侵检测和实时响应分布化，真正实现了分布式检测的思想我们所提出的入侵检测模型是以自治 Agent 为组织单元，其中有三类自治 Agent：入侵检测 Agent(IDA)、数据挖掘 Agent(DMA)、通信服务 Agent(TSA)，如图 4-2 所示

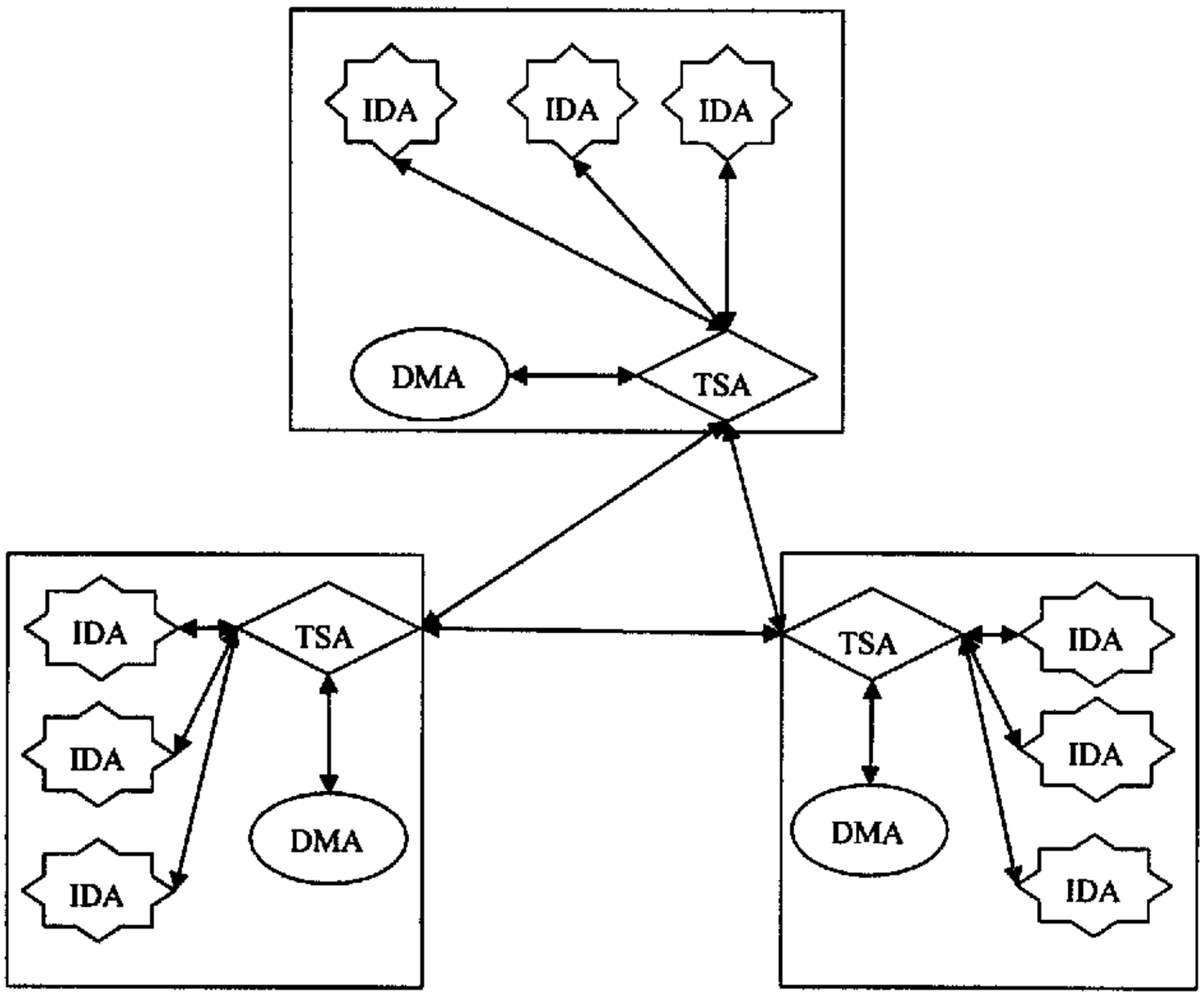

图4-2 多Agent的分布式结构示意图

Figure 4-2 Distributed structure of multi-Agents

4.1.2.2 通信服务代理(TSA)

TSA 是专门用于通信服务[11]的 Agent，它在每台主机上都是唯一的，是主机内和协作主机向 IDA 通信的代理，然后将数据传送给 TSA，TSA 将数据包转送给

目标主机上的 TSA，目标主机上的 TSA 再将数据转送给目的 IDA。

TSA 是数据传送的中介，它记录了与本机所有的 IDA 和协作主机 TSA 的联络方式，可以为数据包提供路由服务它的任务主要是数据的接收和转发，不具有检测能力和控制能力它应该可以识别两种包：广播包和定向数据包广播包是发向除广播源以外的所有 IDA 的，而定向包则是转发给指定的 IDA 如果定向包是发给本地 IDA 的，TSA 就直接转发给它；如果是转发给协作主机上的 IDA 的，就转发给协作主机的 TSA，由协作主机上的 TSA 转发给相应的 IDA。

4.1.2.3 入侵检测代理(IDA)

IDA 是本模型的基本检测单元，它们分布在主机和网络各处，每个 IDA 独立承担一定的检测任务[13]，检测系统或网络安全的一个方面在模型中，各 IDA 有独立的数据源、运行模式和响应方式，各 IDA 之间进行相互协作，对系统和网络用户的异常或可疑行为进行检测不同的 IDA 按照检测环境的不同，采用不同的检测方法和技术。

在本系统中，各 IDA 的行为模式是很相似的，一般都经过以下几个步骤[36, 37](如图 4-3 所示)：

(1)截获系统或网络信息，作日志；

(2)进行审计分析(异常越界检查，并对相应事件进行响应)；

(3)根据规则库进行数据处理，确定可疑度；

(4)与其他 IDA 通信，对可疑级别达到一定程度的事件进行可疑广播；

(5)如果需要将数据传送给其他 IDA，就通过 TSA 将数据送出；

(6)记录用户信息。

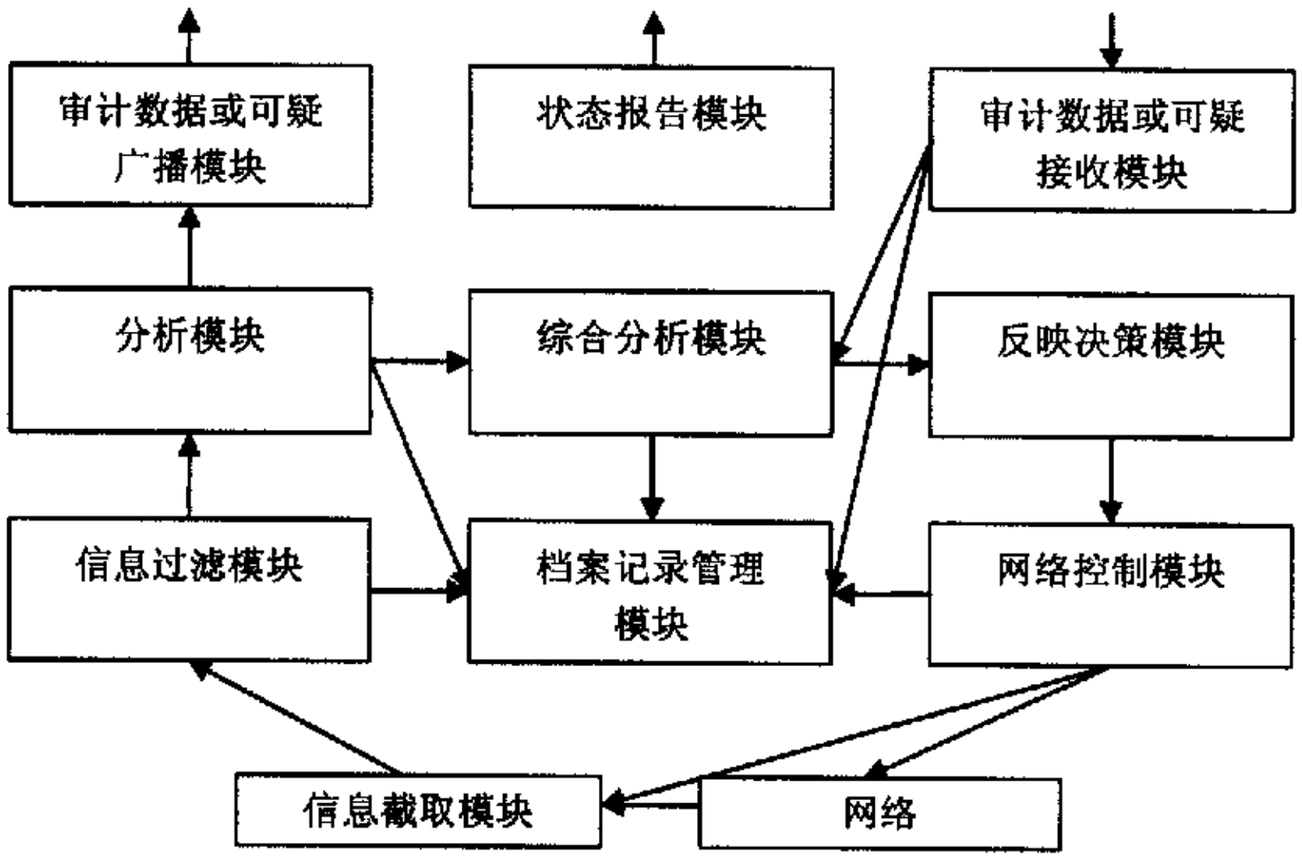

图4-3　IDA检测流程图

Figure 4-3 Flow map of IDA detection

在图 4-3 中，状态报告模块是以其他模块的状态报告作为数据来源的，但为简洁起见，在图中没有标出该模块和其他模块的关系。

IDA 的检测准确性主要取决于检测算法的准确性和规则库的丰富程度，一个比较完全的规则库会提供更多的入侵规则给检测算法，从而有效的提高了入侵检测系统的准确性。这就要求规则库的不断完善和更新，进而我们加入了数据挖掘

代理。

4.1.2.4 数据挖掘代理(DMA)

数据挖掘 Agent 是本模型[26]的规则挖掘单元，数据挖掘 Agent 和检测 Agent 之间通过规则库相连。数据挖掘 Agent 各自采用不同数据挖掘的方法不断的学习，并将学习的结果存入规则库，检测 Agent 通过误用检测，高效率的检测入侵行为。对于一个网段来说，每种 Agent 又分为基于主机的和基于网络的两种。各个主机相互之间通过通信服务 Agent 传递需要的数据和信号。除了数据挖掘 Agent 对规则库不断的更新之外，还可以由网络来学习。网络数据挖掘 Agent 对整个网段的数据进行分析学习，发现对整个网段的入侵。

数据挖掘 Agent 应用数据挖掘算法学习到新的入侵规则后就把该规则加入到规则库[27]，并同时向各个通信服务代理(TSA)发送消息，消息中包括学习出来的新的入侵规则，然后由通信服务代理(TSA)把该规则发送给各个入侵检测 Agent 和其他通信服务代理(TSA)，依次发往各个网络节点，使规则信息得到同步。

数据挖掘 Agent(DMA)执行具体的从网络或主机上提取数据进行入侵检测的功能，也就是一个循环数据采集和数据分析的过程，其一般流程如图 4-4 所示[28]。

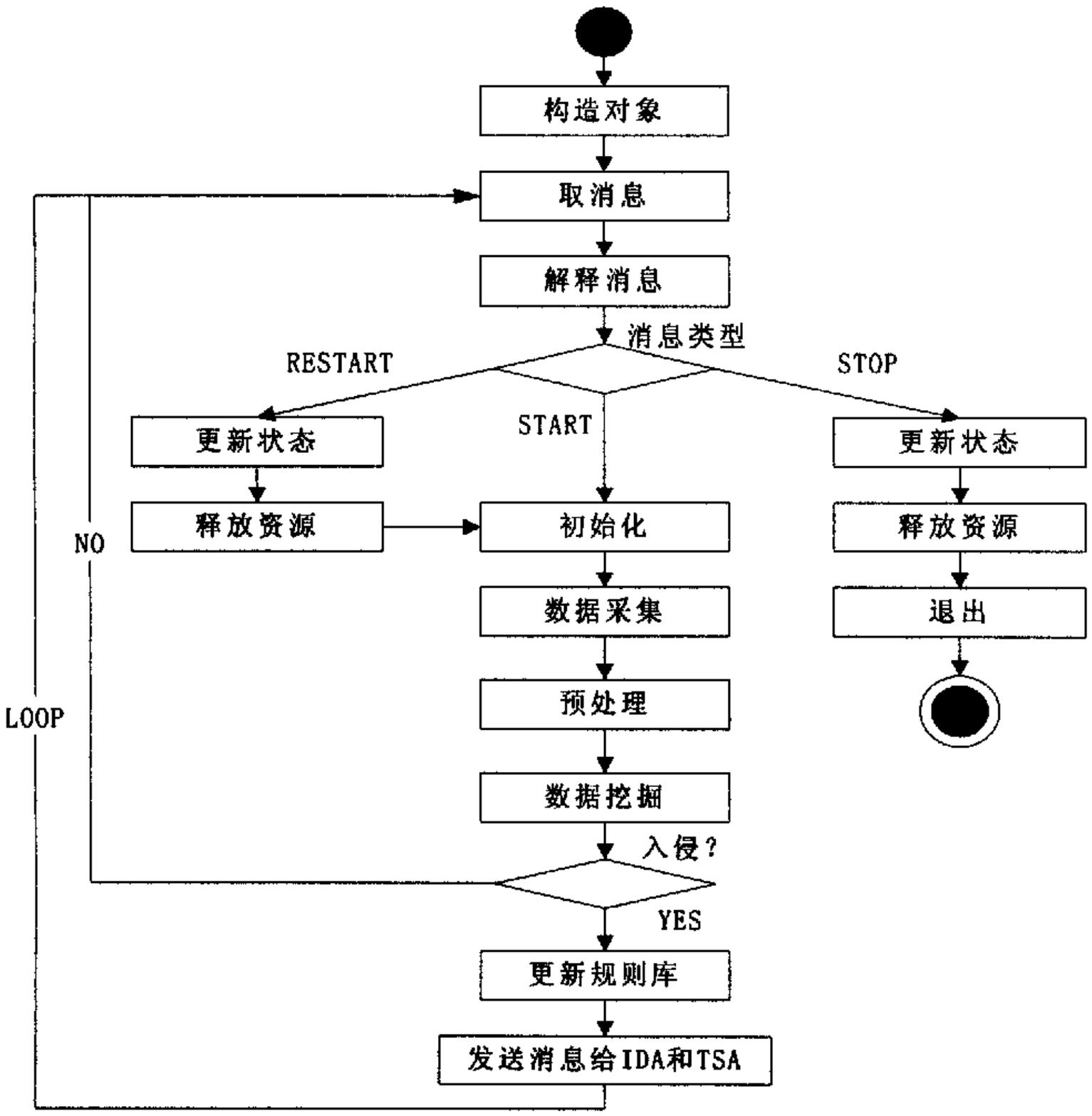

图4-4 DMA工作流程图

Figure 4-4 Flow map of DMA

虽然 DMA 的具体功能有所不同，但 DMA 的行为模式基本一致，只要符合这个

模式的 DMA 都可以纳入 DA_DIDS 的代理体系中。当我们增加新的 DMA 或者修改已有的安全产品，只需继承 DMA 的基类 CDMABase，然后重载相应的数据采集和数据挖掘方法就行了，而不须再考虑 DMA 的运行机制，这就大大增加了系统的灵活性和扩展性。

4.2 DA_DIDS 的运行机制

DA_DIDS 的运行流程如图 4-5 所示：

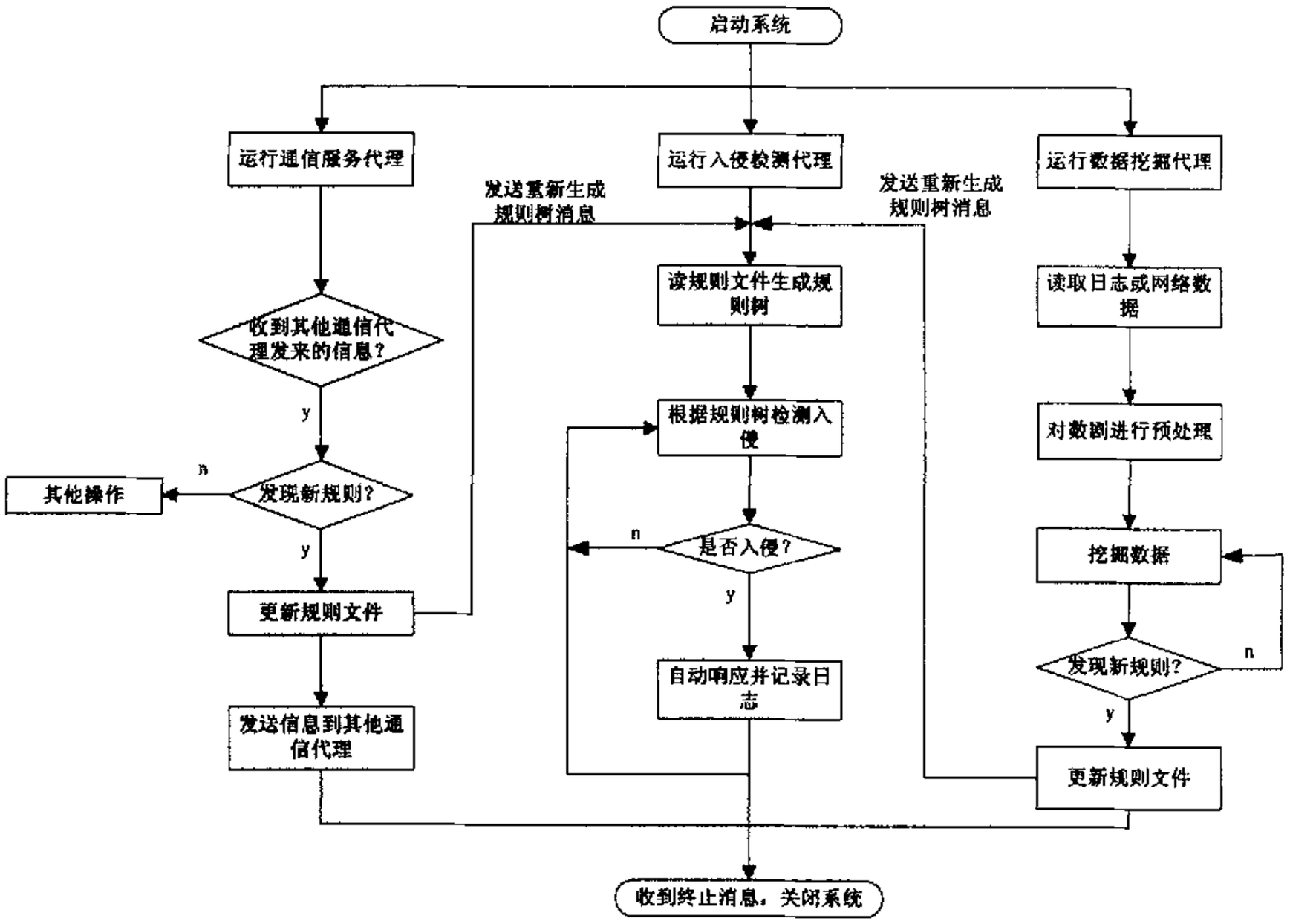

图4-5 DA_DIDS的运行流程

Figure 4-5 Flow map of DA_DIDS

图中 4-5 只描绘了一个入侵检测代理和一个数据挖掘代理，实际系统中可以有多个不同的入侵检测和数据挖掘代理。图 4-5 中重要的部分是规则的匹配和数据挖掘，将分别在下文中介绍。

4.2.1 规则知识库的建立与更新

规则的一般格式定义为(本文采用 snort 的规则格式定义)：

〈规则操作〉〈协议〉〈源主机 IP〉〈源端口〉〈方向操作符〉〈目标主机 IP〉〈目标端口〉(〈规则选项 1：值 1〉；〈规则选项 2：值 2〉；…；〈规则选项 n：值 n〉；)

规则分为规则头(Rule Header)和规则选项(Rule Options) 前后两个部分。规则头包含有匹配后的动作、协议类型、以及 TCP/IP 的四元组(源目的 IP 地址

及源目的端口)。规则选项包含了入侵的特征信息和其它一些信息，由一个或几个选项组合而成，所有主要选项之间是“与”的关系。选项之间可能有一定的依赖关系，选项主要可以分为四类，第一类是数据包相关各种特征的描述选项，比如 content、flags、dsize、ttl 等；第二类是规则本身相关一些说明选项，比如 sid、classtype、rev 等；第三类是规则匹配后的动作选项，比如：msg、resp、react、logto、tag 等；第四类是选项是对某些选项的修饰，比如从属于 content 的 nocase、offset、depth 等。

例如规则：alert tcp any any -> 192.168.100.68 80 (msg:"WEB-IIS ISAPI.ida attempt ; uricontent:".ida?" ; nocase ; dsize:>239 ; classtype:web-application-attack; sid:1243; rev:6;)

在上面所举的例子中，括号左面为规则头，括号中间部分为规则选项，规则选项中冒号前面的部分是关键字(Option Keyword)，关键字根据检测某种入侵和采取某种反应行为的需要进行组合，各选项之间用分号隔开。规则中各因子之间是“与”的关系，只有当各因子都为真是才触发相应的操作。而整个规则知识库中，各条规则之间是“或”的关系，只要一条规则为真，我们就认为有入侵的发生。

规则库作为 IDS 的知识中心，必须在进行检测流程之前建立，而且为了提高检测的速度，规则知识库必须具有合理的内存结构以便常驻内存。在传统的入侵检测系统中其建立过程主要由两步组成：

1、读取规则文件，进行规则的解析；

2、在内存中对规则进行组织，建立规则语法树。

而在我们基于数据挖掘的入侵检测系统中还要再增加一步：

3、把 DMA 对网络和主机数据进行挖掘分析不断的学习出新的规则加入规则文件和规则语法树，使规则库随着时间动态增长。

规则知识库逻辑结构如图 4-6 所示。整个规则知识库按照链表组织，链表按照 TCP/IP 协议类型进行分类，现包括四种规则链表，IP、TCP、UDP 和 ICMP 规则链(IPList，TCPList，UDPList 和 ICMPList)，每个规则链都是一棵规则语法树。所有的规则都按照规则头(Rule Tree Node，RTN)进行组织，由规则头排成主链，而相同规则头的规则选项(Rule Option Tree Node，OTN)构成从链，然后根据规则选项把规则插入到相应的链表中，构成一棵规则语法树，这样每一颗选项结点就对应一条规则。

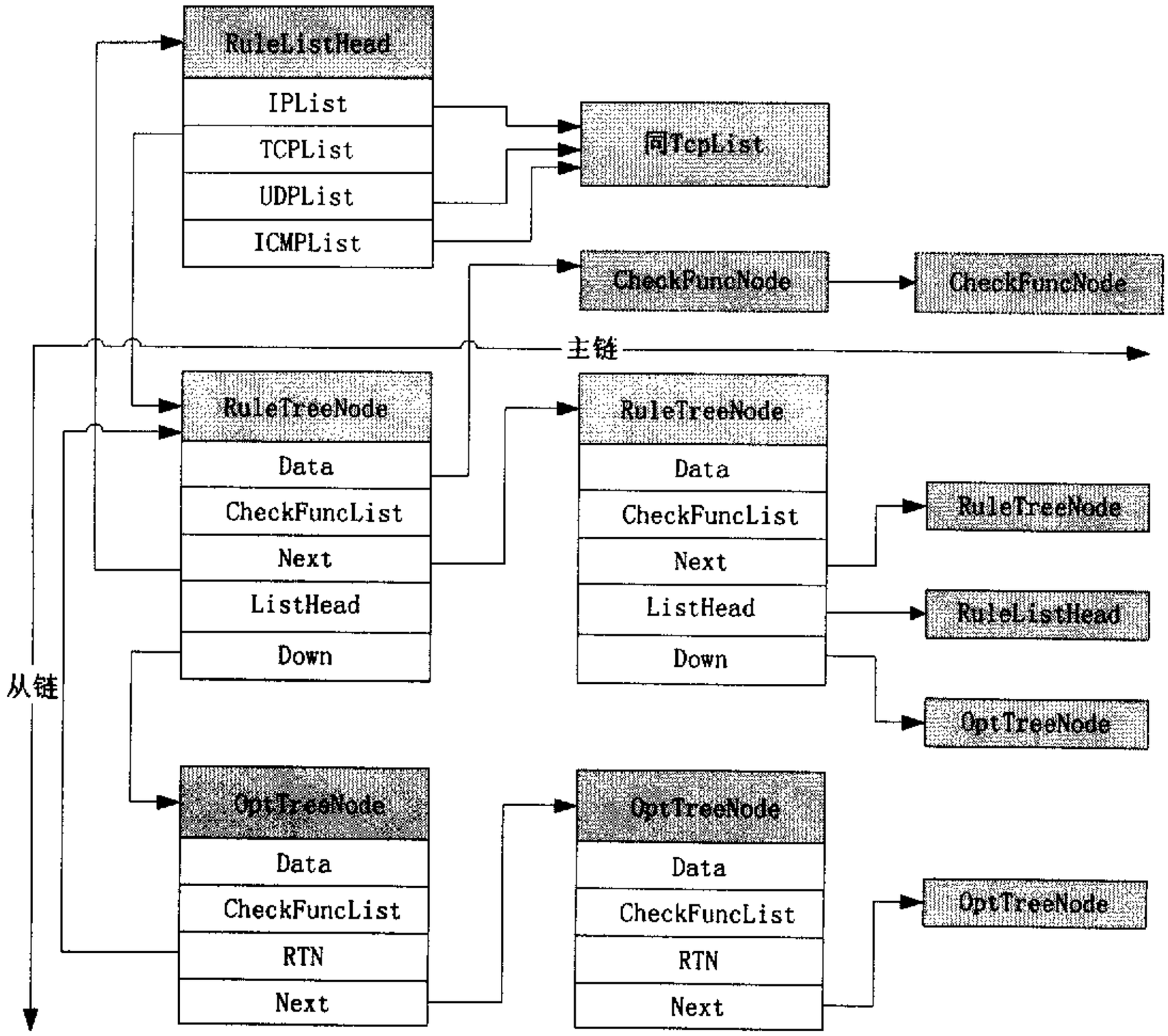

图4-6 规则知识库规则语法树结构图

Figure 4-6 The Structure of Rule Tree

规则语法树的数据结构我们采用了伸展二叉树—SplayTree，SplayTree 是二叉排序树的一个改进，特点是：当任何一个结点被访问时，树就必须重新排序，将被访问的结点移动到根，同时保持树的中序排列顺序不变。SplayTree 使得经常访问的结点离根很近，从而维护了较好的平衡性。

规则语法树的生成过程比较简单，首先读取规则文件，依次取出每一条规则；然后对读取的规则进行解析，并用相应的规则语法表示，建立规则语法树；最后进行一些完善操作，并进行规则语法树的完整性检查。流程图如图 4-7 所示。

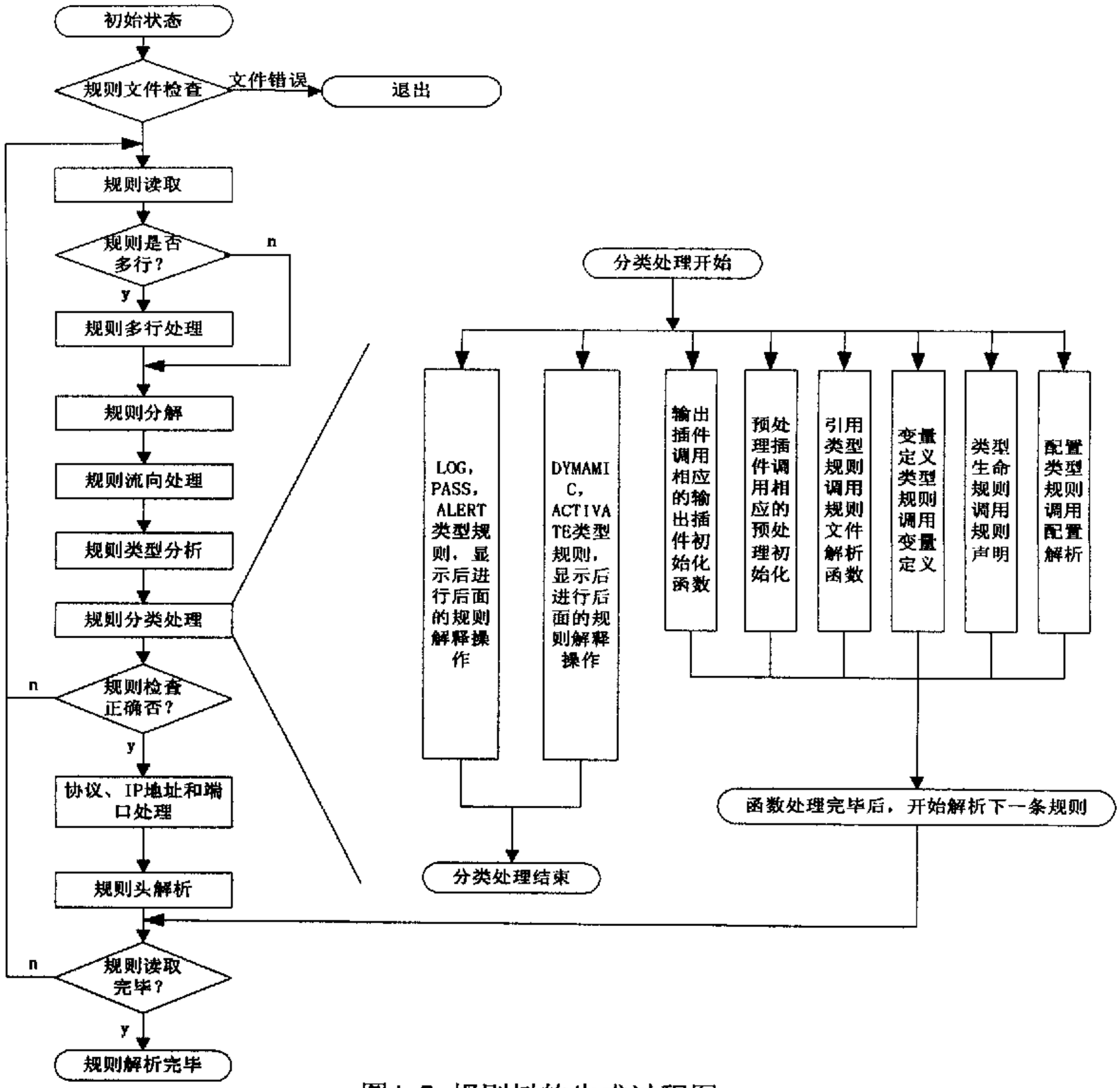

图4-7 规则树的生成过程图

Figure 4-7 creation of rule tree

4.3 数据挖掘代理挖掘规则

4.3.1 数据的采集

入侵检测的第一步就是信息的收集，内容包括系统、网络、数据及用户活动的状态和行为。入侵检测利用的信息很大一部分是来自系统的日志文件和审计信息。入侵者经常在系统日志中留下他们的踪迹，因此充分利用系统的日志文件和审计信息是检测入侵的必要手段。日志中包含发生在系统和网络上的不寻常和不期望活动的证据，这些证据可以指出有人正在入侵或已经成功地入侵了系统。通过查看日志文件和系统产生的审计信息，能够发现成功的入侵或入侵企图。

根据入侵检测的功能，可以把它的功能结构分为两大部分[39, 40]：中心平台和代理服务器。代理服务器是负责从操作系统中采集审计数据，并把审计数据转换成与平台无关的格式后

传送到中心检测平台，或者把中心平台的审计数据需求传送到各个操作中。而中心平台由专家系统知识库和管理员组成，其功能是根据代理服务器采集来的审计数据进行专家分析，产生安全报告。管理员可以向各个主机提供安全管理功能，根据专家系统的分析向各个代理服务器发出审计数据的需求。

在一个部署有入侵检测系统的网络中，网络通信量是极大的，日常的系统日志的数据量非常大。在一个系统正确、合理地配置了审计机制后，便可以源源不断地产生大量的审计数据。所以，在系统中存在大量的日志和审计数据。这些信息有的与入侵有关，有的与入侵无关。由于数据挖掘技术在数据中提取特征与规则方面有非常大的优势，所以我们可以采用数据挖掘技术从那些大量的，甚至是冗余的系统日志与审计数据信息中提取对安全有用的信息。

1、系统日志

在 Unix 中有许多日志，Unix 的日志一般有以下一些内容，如表 4-1。

表 4-1 Unix 的日志说明

Figure 4-1 logs on Unix

日志	说明	日志	说明
acct 或 pacct	记录每个用户使用的命令记录	access_log	主要使用于服务器运行了 NCSA HTTPD，这记录文件会有什么站点连接过你的服务器
aculog	保存着你拨出去的 MODEMS 记录	lastlog	记录了用户最近的 LOGIN 记录和每个用户的最初日的地，有时是
loginlog	记录一此不正常的 LOGIN 记录	messages	记录输出到系统控制台的记录，另外的信息由 syslog 来生成
utmp	记录当前登录到系统中的所有用户这个文件伴随着用户进入和离开系统而不断变化	sulog	记录使用 su 命令的记录
syslog	最重要的日志文件，使用 syslogd 守护程序来获得日志信息:/dev/log 一个 UNIX 域套接字，接受在本地机器上运行的进程所产生的消息/dev/log 一个从 UNIX 内核接受消息的设备 514 端口一个 INTERNET 套接字，接受其他机器通过 UDP 产生的 syslog 消息。	wtmp	记录用户登录和退出事件
		ftp 日志	执行带-l 选项的 ftpd 能够获得记录功能
		http 日志	HTTPD 服务器在日志中记录每一个
		History 日志	这个文件保存了用户最近输入命令
uucp	记录的 UUCP 的信息，可以被本地 UUCP 活动更新，也可有远程站点发起的动作修改，信息包括发出和接受的呼叫，发出的请求，发送	securtiy	记录一些使用 UUCP 系统企图进入限制范围的事例

	者，发送时间和发送主机。		

2、系统审计

在 Unix 操作系统中带有许多审计机制，并且还可以再配置审计工具，因此能够产生大量的审计数据。这些审计数据对于分析评价系统的安全性非常有价值。操作系统审计要记录以下类型的事件：使用识别与认识机制(如注册过程)，将某个客体放入某个用户的地址空间(如打开文件)，从用户地址空间中删除客体、计算机操作员、系统管理员和系统安全员进行的操作及其他所有与安全有关的操作。对于每个记录下来的事件，审计记录应该能够标识出事件发生的时间，触发这一事件的用户、事件的类型及事件成功与否等等。对于识别与认证事件，审计记录应该能记录下事件发生的源地点(如终端标识符)；对于将某个客体引入某个用户的地址空间及从用户地址空间中删除客体的事件，审计记录应该记录下客体名及客体的安全级等信息。

另外，还可以对系统另外配置一些其他的审计工具，譬如 TCP Wrapper, Sendmail, netlog 等等。这些实用程序使用特定的日志文件产生它们的信息，我们可以收集这些使用程序的输出，结合其他日志描述出一幅系统的真实状态图。

4.3.2 对采集数据选择相关的审计日志信息并进行预处理

数据挖掘中的预处理主要是接受并理解用户的发现需求，确定发现任务，抽取并发现与任务有关的知识源，根据背景知识中的约束性规则对数据进行合法性检查，通过清理和归约等操作，生成供数据挖掘核心使用的目标数据，即知识基。知识基是原始数据库经数据汇集处理后得到的二维表，纵向为属性(Attributes 或 fields)，横向为 Tables 或 Records)。它汇集了原始数据库中与发现任务相关的所有数据的总体特征，是知识发现状态空间的基底，也可以认为是最初始的知识模版。系统的审计日志信息非常庞大，并存在杂乱性，重复性和不完整性等问题。由于原始数据是从各个代理服务器中采集后送到中心检测平台的，而各个代理服务器的审计机制的配置并不完全相同，所产生的审计日志信息存在一些差异，因此有些数据就显得杂乱无章；重复性指对于同一个客观事物在系统中存在其两个或两个以上完全相同的物理描述。在 Unix 操作系统中的审计日志信息是存在许多重复和冗余现象的。例如，在 Unix 系统中，记录用户登录的日志有 Lastlog,UTMP, WTMP。这 3 个日志记录的内容有所区别，Lastlog 是记录每个用户最近一次登录时间和每个用户的最初目的地，UTMP 记录以前登录到系统中的所有用户，而 WTMP 文件记录用户登录和退出事件。从这些可以看出，这 3 个日志文件中是存在许多相同项目的，这样就带来了数据的重复和冗余问题。我们可以通过数据集成、数据清理、数据简化等方法对系统的审计日志信息进行预处理。

数据集成主要是将多文件中的异构数据进行合并处理，解决语义的模糊性。该部分主要涉及数据的选择，数据的冲突性以及不一致性数据的处理问题。

数据清理要去除源数据审计日志信息数据集中的噪声数据和无关数据，处理遗漏数据和清洗脏数据，去除空白数据域和知识背景上的白噪声，考虑审计日志信息的时间变化和它们的数据变化。主要是对重复数据和缺值数据进行处理，去除重复数据记录，填补缺省数据。

系统审计日志中有些数据属性对提取安全规则没有影响，这些属性的加入会

大大影响数据挖掘效率，甚至还可能导致数据挖掘结果的偏差。因此，有效地对数据进行简化是很有必要的。数据简化是在对发现任务和数据本身内容理解的前提下最大限度地精简数据量。主要通过两条途径[41, 42]:属性选择和数据抽样。分别对系统审计日志信息中的数据的属性和记录进行简化。属性选择包括根据需要提取的安全主题对数据的属性进行剪枝、并枝等相关操作。剪枝就是去除对提取安全规则没有贡献或贡献率很低的属性值。并值就是把相近的属性进行综合归并处理。

数据抽样是对审计日志的数据记录之间进行相关性分析，用那些有代表性的表达入侵特征的记录组合来表达整个审计日志信息，从而减少了记录数目，提高了挖掘效率。

4.3.3 对预处理后的数据进行挖掘

从上面可以看出，系统中的审计日志信息的数据量非常的庞大，为了从这些庞大的信息中提取出对安全有关的信息，我们可以采取数据挖掘的相关技术。入侵检测中对系统日志信息进行挖掘的过程可以分为如下阶段[43]：采集选择，数据预处理、挖掘。如图 3-3 所示。

数据挖掘算法的种类很多，本章重点介绍将在入侵检测系统的数据挖掘代理(DMA)中所使用数据挖掘算法以及相关的改进，包括聚类分析[25]和关联规则两大类。

聚类分析是一种经常被用于入侵检测的数据挖掘技术。由于越来越多的入侵是跨越时间和空间的，这种技术就可以发现那些看起来不相关的网络行为中隐藏的攻击。我们把相关的数据聚积起来，这样也可以适用于异常行为的发现。聚类规则集是对异常检测模型关联规则集的学习训练过程，从而使得到的检测模型更准确更可信。它的基本思想是初始化聚类规则集，然后在采集了一定量的网络数据后对这些网络包进行挖掘得到一些规则，把得到的这些规则归并到聚类规则集中，将聚类规则集更新。对于新产生的规则集中的每条规则，在聚类规则中寻找相匹配的规则。如果在聚类规则集中找到一条相匹配的规则，那么将聚类规则集中这条规则的计数器 count 加 1，并且使用加权平均的方法更新聚类规则集中该规则的支持度和可信度。否则，如果没有找到一条匹配的规则，就将这条规则加入到聚类规则集中，并将该条规则的计数器 count 置为 1。这样训练聚类规则集若干次，直到聚类规则集稳定为止(即直到没有或很少有新的规则加入)。具体来说是对聚类规则集进行压缩，删除 count 小于用户指定的最小 count 值的那些规则。删剪后的聚类规则集作为检测过程中使用的规则集，建立规则库。

一个能产生高质量聚类的好聚类(Good Clustering)算法必须满足下面两个条件：

1、类内(intra-class)数据或对象的相似性最强；

2、类间(inter-class)数据或对象的相似性最弱。

聚类质量的高低通常取决于聚类算法所使用的相似性测量的方法和实现方式，同时也取决于该算法能否发现部分或全部隐藏的模式。

4.3.3.1 聚类分析中的数据类型

许多基于内存的聚类算法选择如下两种有代表性的数据结构：

● 数据矩阵(Data Matrix，或称为对象与变量结构)：它用 p 个变量(也称为度量或属性)来表现 n 个对象，例如年龄、身高、体重、性别、种族等属性来表现对象“人”。这种数据结构是关系表的形式，或者看成 $n\times p$ (n 个对象×p 个变量)的矩阵。

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix} \tag{4-1}$$

● 相异度矩阵(Dissimilarity Matrix，或称为“对象——对象“结构)：存储 n 个对象两两之间的近似性，表现形式是一个 $n\times n$ 维的矩阵。

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix} \tag{4-2}$$

在这里 d(i ，j)表示对象 i 和 j 之间的相异性的量化表示，通常它是一个非负的数值有 d(i ，j)= d(j， i)，d(i，i)=0，并且当对象 i 和 j 越相似或接近，其值越接近 0；两个对象越不同，其值越大。

数据矩阵经常被称为二模矩阵(two-mode)，而相异度矩阵被称为单模(one-mode)矩阵，这是因为前者的行和列代表不同的实体，后者的行和列代表相同的实体，许多聚类算法以相异度矩阵为基础。如果数据是用数据矩阵的形式表示的，在使用该类算法之前要将其化为相异度矩阵。

4.3.3.2 相似性的测度和聚类准则

为了将数据或对象集合划分成不同类别，必须定义相似性的测度来度量同一类别之间数据的相似性和不属于同一类别数据的差异性。同时要考虑到数据的多个属性使用的是不同的度量单位，这些将直接的影响到聚类分析的结果，所以在计算数据的相似性之前先要进行数据的标准化。针对不同的数据属性类型有不同的标准化方法，这里主要介绍的是对于数值属性的标准化方法，因为入侵检测数据的属性大部分都是数值属性(其他的非数值属性可以在数据离散化的过程中转化为数值或是用关联规则来分析)。

对于一个给定 n 维(属性)数据 f，主要有两种标准化方法：

（1） 平均绝对误差 S_f

$$S_f = \frac{1}{n}\sum_{i=1}^{n}\left|x_{if} - m_f\right| \tag{4-3}$$

这里 x_{if} 表示的是 f 的第 i 个属性，m_f 是 f 的平均值，即

$$m_f = \frac{1}{n}\sum_{i=1}^{n} x_{if} \tag{4-4}$$

(2)标准化度量值 z_{if}

$$z_f = \frac{x_{if} - m_f}{s_f} \tag{4-5}$$

这个平均的绝对误差 s_f 比标准差 σ_f 对于孤立点具有更好的鲁棒性。在计算平均绝对偏差时，属性值与平均值的偏差 $\left|x_{if} - m_f\right|$ 没有平方，因此孤立点的影响在一定程度上被减小了。

数据标准化处理以后就可以进行属性值的相似性测量，通常是计算对象间的距离。对于 n 维向量 x_i 和 x_j，有以下几种距离函数：

(1)欧几里得距离：

$$D(x_i, x_j) = \left|x_i - x_j\right| = \sqrt{\sum_{k=1}^{n}\left(x_{ik} - x_{jk}\right)^2} \tag{4-6}$$

(2)曼哈顿距离

$$D(x_i, x_j) = \sum_{k=1}^{n}\left|x_{ik} - x_{jk}\right| \tag{4-7}$$

(3)一般化的明氏(Minkowaki)距离

$$D_m(x_i, x_j) = \left[\sum_{k=1}^{n}\left(x_{ik} - x_{jk}\right)^m\right]^{1/m} \tag{4-8}$$

当 m=2 时，明氏距离 D_2 即为欧氏距离；当 m=1 时，明氏距离 D_1 即为曼哈顿距离；

(4)角度相似性函数

$$s(x_i, x_j) = \frac{x_i' x_j}{\|x_i\| \bullet \|x_j\|} \tag{4-9}$$

表示的是向量x_i和x_j之间夹角的余弦，也是x_i的单位向量$x_i/\|x_i\|$与x_j的单位向量$x_j/\|x_j\|$之间的点积。

夹角余弦的测度反映了几何上相似性的特征，它对于坐标系的旋转和放大缩小是不变的，但对位移和一般的线性变换并不具有不变性的性质。当特征的取值仅为(0，1)时(此时称为布尔属性值)，夹角余弦度量具有特别的含义。如果x_i和x_j向量的每一个分量(即属性)都只能是0或1，当其第k个特征分量为1值，认为该向量具有第k个特征；如果为0，则无此特征分量。这样$x_i' x_j$的值等于x_i和x_j二向量共有的特征数目。

$$\|x_i\| \bullet \|x_j\| = \sqrt{(x_i' x_j)(x_j' x_i)} \tag{4-10}$$

x_i中具有特征数目和x_j中具有特征数目的几何平均，所以在0和1的二值情况，$S(x_i, x_j)$等于x_i和x_j中具有共同特证数目的相似性测度。

(5)Tanimoto测度函数(适用二值情况)

$$S(x_i, x_j) = \frac{x_i' x_j}{x_i' x_i + x_j' x_j - x_i' x_j} = \frac{x_i \text{和} x_j \text{中共有的特征数目}}{x_i \text{或} x_j \text{中占有的特征数目之总数}} \tag{4-11}$$

例如。$x_i = (010110)'$，$x_j = (001101)'$

得：$x_i' x_j = 1$，$x_i' x_i = 3$，$x_j' x_j = 3$

$$S(x_i, x_j) = \frac{1}{3+3-1} = \frac{1}{5}$$

数值属性用到的主要是欧几里得距离和测量二值属性的测度函数，对于欧几里得距离和曼哈顿距离满足以下条件：

1) $D(x_i, x_j) \geq 0$：距离是一个非负数值；

2) $D(x_i, x_i) = 0$：对象与自身的距离是零；

3) $D(x_i, x_j) = D(x_j, x_i)$：距离函数具有对称性；

4) $D(x_i, x_j) \leq D(x_i, x_k) + D(x_k, x_j)$：$x_i$到$x_j$的距离不会大于$x_i$到$x_k$和$x_k$到$x_j$的距离之和(三角不等式)。

有了相似性测量函数，下一步要确定的是采用的聚类准则。聚类准则是聚类分析算法的关键，通常有两种确定方式：

1、试探方式：凭主观和经验，针对实际问题定义一种相似性测度的阈值，然后按最近邻规则指定某些对象属于某一聚类。例如使用欧氏距离，它反映的是对象之间的近邻性，在将一个对象分到两个类别中的一个时，必须规定一距离测度的阈值作为聚类的判别准则。

2、聚类准则函数法：由于聚类是将对象进行组合分类以使类别可分离性最大，因此聚类准则应是反映类别间相似性或相异性的函数。但类别是由一个个对象所组成，所以一般说来，类别的可分离性与对象的相异性直接有关。这样，定义一聚类准则函数 J，应是对象集合 N 和聚类类别 $\{x\}$ 和聚类类别 $\{S_j, j=1,2,...c\}$ 的函数。该过程使聚类分析转化为寻找准则函数极值的最优化问题。一种常用的指标是误差的平方和，即

$$J=\sum_{j=1}^{c}\sum_{x\in S_j}\left\|x-m_j\right\|^2 \tag{4-12}$$

式中 c 为聚类类别的数目，$m_j=\frac{1}{N_j}\sum_{x\in S_j}x$ 为属于 S_j 集的对象均值向量，N_j 为 S_j 中的对象个数，在这以均值向量 m_j 为 S_j 中对象的代表。

上式表明，J 代表了分属于 c 个聚类类别的全部对象与其相应类别之间的误差平方和。使 J 值极小的聚类就是我们的目的。这种类型的聚类通常称为最小方差划分，它适用于各类对象密集且数日相差不多，而不同类间的对象又明显分开的情况。当不同类中的对象数目相差很大时，采用误差平方和作为准则函数，有时还可能把对象多的一类分析为二，因为这样分拆其误差平方和的数值会更小，这样就会造成错误聚类，此时就须采用别的准则函数。

4.3.3.3 聚类分析算法的分类

聚类分析算法取决于数据的类型、聚类的目的和应用。大体上，主要的聚类算法可以划分为以下几类：

(1)划分方法(Partioning Method)：给定一个 n 个对象或元组的数据库，一个划分方法构建数据的 k 个划分，每个划分表示一个类，并且 k≤n。也就是说，它将数据划分为 k 个组，同时满足以下的要求：(i)每个组至少包括一个对象；(ii)每个对象必须属于且只属于一个组。

给定要构建的划分的数目 k，划分方法首先创建一个初试划分。然后采用一种迭代的重定位技术，尝试通过对象在划分间的移动来改进划分。一个好的划分的一般准则是：在同一个类中的对象尽可能相近或相关，而不同类中的对象之间尽可能远离或不同。

为了达到全局最优，基于划分的聚类会要求穷举所有尽可能的划分。实际上，绝大多数应用采用的是以下两个比较流行的启发式方法：(i)k 均值算法，在该算法中，每个类用该类中对象的均值来表示；(ii)k 中心点算法，在该算法中，每个类用接近聚类中心的一个对象来表示。这些启发式聚类算法对在中小规模的数据库中发现球状类很适用。为了大规模的数据集进行聚类以及处理复杂形状的类别，基于划分的方法需要进一步的扩展。

(2)层次的方法(Hierarchical Method)：层次的方法对给定的数据对象集合进行层次的分解。根据层次的分解如何形成，层次的方法可以分为凝聚的和分裂的。凝聚的方法，也称为自底向上的方法，一开始将每个对象作为单独的一个组，然后相继地合并相近的对象或组，直到所有的组合并成一个(层次的最上层)，或者达到一个终止条件。分裂的方法，也称为自顶向下的方法，一开始将所有的对象置于一个类中，在迭代的每一步中，一个类被分裂为更小的类，知道最终每个对象在单独的一个类中，或者达到一个终止条件。

层次的方法缺陷在于，一旦一个步骤(合并或分裂)完成，它就不能被取消。这个严格的规定是有用的，由于不用担心组合数目的不用选择，计算代价会较小。但是，该技术的一个主要问题是它不能更正错误的决定。有两种方法可以改进层次聚类的结果：(i)在每层划分中，仔细分析对象间的连接；(ii)综合层次凝聚和迭代的重定位方法，首先采用自底向上的层次算法，然后用迭代的重定位来改进结果

(3)基于密度的方法(Density-based Method)：绝大多数划分方法基于对象之间的距离进行聚类，这样的方法只能发现球状的累，而在发现任意形状的类上遇到了困难。随之提出了基于密度的另一类方法，其主要思想是：只要临近区域的密度(对象或数据点的数目)超过某个阈值就继续聚类。也就是说，对给定类中的每个数据点，在一个给定范围的区域中必须至少包含某个数目的点。这样的方法可以用来过虑噪声孤立点数据，发现任意形状的类。

DBSCAN 是一个有代表性的基于密度的方法，它根据一个密度阈值来控制类的增长。OPTICS 则是另一个基于密度的方法，它为自动的和交互的聚类分析计算一个聚类顺序。

(4)基于网格的方法(Grid—based Method)。基于网格的方法把对象空间量化为有限数目的单元，形成一个网络结构。所有的聚类都是在这个网络结构(即量化的空间)上进行。这种方法的主要优点是它的处理速度很快，其处理时间独立于数据对象的数目，只与量化空间中每一单元的数目有关。

(5)基于模型的方法(Model—based Method)：基于模型的方法为每个类假定了一个模型，寻找数据对给定模型的最佳拟合。一个基于模型的算法可能通过构建反映数据点空间分布的密度函数来定位聚类。它也基于标准的统计数字自动决定聚类的数目，考虑噪声数据和孤立点，从而产生健壮的聚类算法。

一些聚类算法集成了多种聚类方法的思想，所以有时将某个给定的算法划分为属于某类聚类算法是很困难的。此外，某些应用可能有特定的聚类标准，要求综合多个聚类技术。

4.3.3.4 几种聚类算法

根据检测到的入侵数据(主要是网络数据流和系统日志文件)的特点，在入侵检测系统中使用到的聚类算法主要是划分的方法，其基本思想是：在给定 n 个对象和聚类数目 k 后，先选择若干(k 个)样本点做为聚类中心；再按某种聚类准则(通常采用最小距离准则)使各个样本点向中心聚集，从而得到初始分类；然后，判断分类是否合理，如果不合理，就修改分类，……，以此反复修改聚类的迭代运算，直到合理为止。在这类方法中，最著名和最常用的算法是 k-均值算法、k 中心点算法和它们的变种。

- k-均值(k-means)算法

该算法基于使聚类性能指标最小化的原则，通常使用的聚类准则函数是聚类集中的每个样本点(数据或对象)到该类中心的聚类平方之和，并使它最小化，计算的步骤为：

(i)选 k 个初始聚类中心：$z_1(l)$，$z_2(l)$，…，$z_k(l)$，括号内的序号为寻找聚类中心的迭代运算的次序号。聚类中心的向量值可以任意设定，例如可用开始 k 样本点作为初始聚类中心。

(ii)逐个将需分类的样本$\{x\}$按最小距离原则分配给聚类中心的某一个$z_j(l)$假如 i＝j 时，$D_j(l)=\min\left\{\left\|x-z_j(l)\right\|,\ i=1,2,...k\right\}$，则$x\in S_j(l)$，其中$l$为迭代运算次序号，第一次选代则$l=1$，$S_j$表示第 j 个聚类，其聚类中心为$z_j$。

(iii)计算各个聚类中心的新的向量值：

$$z_j(l+1), j=1,2,...k \tag{4-13}$$

求各聚类域中所包含样本的均值向量，即

$$z_j(l+1)=\frac{1}{N_j}\sum_{x\in S_j(l)}x, j=1,2,...,k \tag{4-14}$$

式中N_j是第 j 个聚类域S_j中所包含的样本个数。以均值向量为新的聚类中心，可以使聚类准则函数

$$J_j=\sum_{x\in S_j(l)}\left\|x-z_j(l+1)\right\| \tag{4-15}$$

最小，其中 j=1，2，…k。这一步要分别计算 k 个聚类中的样本均值向量，k 均值　　算法由此得名。

(iv) 如果$z_j(l+1)\neq z_j(l), j=1,2,...k$，则$l=l+1$，回到(ii)，将样本逐个重新分类，重复迭代计算。$z_j(l+1)=z_j(l), j=1,2,...k$，则算法收敛，计算完毕。

k-均值算法尝试找出使平方误差函数值最小的 k 个聚类。当样本数据是密集的，且类与类之间区别特别好的时候，该算法的效果较好。对处理大数据集，该算法是相对可伸缩和高效率的，因为它的复杂度是 0(nkl)，其中 n 表示所有样本点的个数，k 是聚类数，1 是迭代次数，通常 k<<n 且 l<<n。

k-均值算法经常以局部最优结束。K-均值算法需要给出类平均值，因此它不适合处理有分类属性的数据。K-均值算法生成的聚类数是预先给定的，不能动态的添加新的聚类，也是该算法的一个缺点。此外，该算法对差别很大带有孤立点数据的类的聚类效果不是很好，因此实际中使用时需要对该算法加以改进。

- k-中心点(k-medoids)算法

k-均值算法对孤立点是敏感的，若是样本点属性值极大的情况将影响整个聚类域分布。为了消除这个敏感性，可以采用聚类中的一个样本点而不是聚类域中的样本均值作为聚类的中心，这个样本点就称为中心点(medoids)。这种算法还

是基于使聚类性能指标最小化的原则，不过使用的是样本作为中心。具体步骤：

(i)任意选取 k 个样本点作为中心点O_1，O_2，…O_k；

(ii)将余下的样本点分到各个类中去(根据与中心点最相近的原则)；

(iii)随机选取一个非中心样本点Q_{random}；

(iv)计算用Q_{random}代替Q_j(中心点)的代价函数 S；

(vi)重复第 M 步直到中心点不再发生变化为止。

聚类结果的质量用一个代价函数来估算，该函数度量样本与中心点之间的平均相异度。为了判定一个非中心样本Q_{random}是否是当前样本中心点的好的替代，对于每一个非中心点 p，下面四种情况被考虑：

a. p 当前隶属于中心点Q_j。如果Q_j被 Q_{random}替代作为中心点，且 p 离一个中心点Q_i最近，i≠ j，那么产被重新分配给Q_i；

b. p 当前隶属于中心点Q_j。如果Q_j被Q_{random}替代作为中心点，且产离一个中心点Q_{random}最近，那么产被重新分配给Q_{random}；

c. P 当前隶属于中心点 Q_i， i≠ j。如果Q_j被Q_{random}替代作为中心点，且 P 仍然离Q_i最近，那么不发生变化；

d. p 当前隶属于中心点Q_i，i≠ j。如果Q_j被Q_{random}替代作为中心点，且 p 离一个中心点Q_{random}。最近，那么 p 被重新分配给Q_{random}。

每当重新分配发生的时候，平方误差 E 所产生的差别对代价函数是有影响的，因此代价函数是计算一个当前的中心点Q_i被Q_{random}替换带来的差别，若是代价减小，Q_i被Q_{random}替代作为中心点，若是代价增大，则当前中心点Q_i保持不变；算法迭代的总趋势是使代价函数不断减小。

与 k-均值算法相比，当存在噪声和孤立点数据的时候，k-中心点算法的聚类效果相对要好一些，这是因为中心点不象平均值那样容易受大数据的影响。但是 k-中心点算法的执行代价比较高。

- k-原型(k-prototypes)算法

k-均值算法是聚类大数据集的有效的和广泛采用的一种聚类算法，然而，它所处理的数据对象只限于数值(numeric)类型的数据。大多数实际的数据库和大的数据集不仅包括数值类型的数据而且包括大量的非数值类型数据，如二值(binary)类型和分类(Categorical)类型数据。Huang 提出了 k-模式小 modes)算法和 k-原型算法来解决具有分类类型数据聚类和具有数值和分类混合类型数据

的聚类问题。k-模式算法采用匹配差异度处理分类数据对象，以模式代替聚类的均值，用基于频率的方法进行聚类过程中的模式的更新来使聚类代价函数达到最小。K-原型综合了 k-均值算法和 k-模式算法的结果，在即考虑数值属性也考虑分类属性的基础上重新定义了差异度：假定 S_r 是由欧式距离定义的数值属性上的差异侧度，S_c 是由两个数据对象的类型的不匹配的个数所定义的分类属性上的差异侧度，则新的差异度定义为：$S_r+\gamma S_c$，其中 γ 是权值系数，这样的定义能够控制哪一种属性在聚类过程中占主导地位。如对于只具有数值属性的数据，它就相当于 k-均值算法 k-原型聚类算法数学推导及其步骤：假定 $X=\{X_1,X_2,\ldots,X_n\}$ 是一组数据元组，其中 $X_i=\lfloor x_{i1},x_{i2},\ldots,x_{i,}\rfloor$ 表示具有 m 个属性值的数据对象。设 k 是一个正整数，对 X 聚类的目的就是要将 X 中的 n 个对象划分到 k 个不同的类别中。一般用使以下代价函数最小作为聚类的准则：

$$E=\sum_{i=1}^{k}\sum_{j=1}^{n}y_{il}d(X_i,Q_l) \tag{4—16}$$

这里，$Q_l=\lfloor q_{l1},q_{l2},\ldots,q_{lm}\rfloor$ 是能够代表聚类 l 的向量或称作原型，$y_{il}\in\{0,1\}$ 是划分矩阵 $y_{n\times l}$ 的一个元素，$y_{il}=1$ 表示数据对象 x_i 属于聚类 l，否则 x_i 不属于聚类 l。d 是差异测度，可用欧氏距离表示。式(4－16)中的 $E_l=\sum_{i=1}^{n}y_{il}d(X_i,Q_l)$ 是将 X 划分到聚类 l 中的代价，即聚类 l 中所有数据对象到它的原型 Q_l 的距离之和。当

$$q_{lj}=\frac{1}{n_l}\sum_{i=1}^{n}y_{il}x_{ij},j=1,2,\ldots,m \tag{4-17}$$

时，E_l 达到最小，其中 $n_l=\sum_{i=1}^{n}y_{il}$ 是聚类 l 中的数据对象个数。这里利用了 k-均值聚类的方法。

如果 X 中的数据对象有分类属性，定义差异测度：

$$d(X_i,Q_l)=\sum_{j=1}^{m_r}\left(x_{ij}^{r}-q_{lj}^{r}\right)^2+y_l\sum_{j=1}^{m_s}\delta\left(x_{ij}^{c},q_{lj}^{c}\right) \tag{4-18}$$

其中，当 p=q 时，$\delta(p,q)=0$，否则，$\delta(p,q)=1$ 。x_{ij}^{r} 和 q_{lj}^{r} 分别是数据对象 X_i 和聚类 l 的原型 Q_l 的数值属性的取值，而 x_{ij}^{c} 和 q_{lj}^{c} 是分类属性的取值。m_r

和m_c别是数值属性和分类属性的个数。γ_l是聚类l的分类属性的权值。如果对所有聚类，γ_l取同一值，则用γ表示。

$$E_l = \sum_{i=1}^{n} y_{il} \sum_{j=1}^{m_r} \left(x_{ij}^r - q_{lj}^r\right)^2 + \gamma_l \sum_{i=1}^{n} y_{il} \sum_{j=1}^{m_l} \delta\left(x_{ij}^c, q_{lj}^c\right) = E_l^r + E_l^c \qquad (4\text{-}19)$$

其中，E_l^r是基于聚类l中所有数据对象数值属性的代价。设c_j是分类属性j上的所有取值集合，$p(c_j \in C_j | l)$是聚类l中数据对象在分类属性j上取值为c_j的频率，则：

$$E_l^c = \gamma_l \sum_{j=1}^{m} n_l (1 - p(q_{lj}^c \in C_j | l)) \qquad (4\text{-}20)$$

n_l是聚类l中的数据对象个数。要使E_l^c最小，当且仅当$p\left(q_{lj}^c \in C_j | l\right) \geq p(c_j \in C_j | l)$，$q_{lj}^c \neq c_j$，$j = 1,2,...m$即，对一个指定的聚类$l$，其原型的分类属性只有取聚类中数据对象的分类属性中出现最频繁的值才能使代价最小。

聚类一个具有数值属性和分类属性的数据集的代价函数可写作：

$$E = \sum_{l=1}^{k} \left(E_l^r + E_l^c\right) = \sum_{l=1}^{k} E_l^r + \sum_{l=1}^{k} E_l^c = E^r + E^c \qquad (4\text{-}21)$$

每个聚类原型的数值属性可由k-均值确定，其分类属性可由上述原则选取。

综上所述，k-原型聚类算法步骤描述如下：

(1)从数据集中选取k个初始聚类原型；

(2)按照上述定义的差异度最小原则将数据集中的数据对象划分到离它最近的聚类原型所代表的聚类中；

(3)对于每个聚类，重新计算聚类原型；

(4)计算每个数据对象对于新的聚类原型的差异度，如果离一个数据对象“最近”的聚类原型不是当前数据对象所属聚类的原型，则重新分配这两个聚类的对象；

重复(3)、(4)直到各个聚类中不再有数据对象发生变化。

4.4 基于数据挖掘的分布式入侵检测模型的优点

(1)独立性。IDA是独立运行的程序实体，可以独立开发在放入具体运行环境前，可以进行独立测试。

(2)具有学习性。DMA具有很强的自学习能力，可以用不同的学习算法来对网络或本地数据进行特征分析，可以进行有预见性的入侵分析，并把学习出来的入侵特征及时加入到规则库，使得规则库不断得到动态增加。

(2)灵活性。 IDA 可以独立地启动和停止，也可以进行动态配置，而不影响其他 IDA 的正常运行要收集新数据或检测新类型的入侵，可以通过对原 IDA 进行重新配置或增加 IDA 来实现。
(3)系统可扩充性好。无论是增加检测主机，还是在主机中增加 IDA 都简单、方便。
(4)错误扩散小。如果某个 IDA 出现问题或受到破坏，那么，仅仅与该 IDA 相关的检测部分失效，限制在最小的范围内。
(5)数据来源不受限制。不同的 IDA 可以选用不同的数据源由于各 IDA 是独立实现的，那么数据来源就可以采用多种形式：审计数据、检查系统配置、捕获网络包或其他合适的来源。
(6)兼容性。该模型可以既包含基于主机的 IDA，又包含基于网络的 IDA，超越了传统入侵检测模型的界限。
(7)与平台和开发语言无关。由于 IDA 是独立的，它们可以分别开发，而且可以基于不同平台使用不同的语言开发，只要遵循统一的通信协议和通信格式就可以了。
(8)协作性。虽然每个 IDA 检测的只是主机或网络安全的一个方面，甚至可能是简单的命令审查，但由于 IDA 之间可以交换信息，就可以在一些 IDA 中产生很复杂的结果。

4.5 小结

本章对基于数据挖掘的分布式入侵检测系统的结构做了详细的说明，并讲解了 DA_DIDS 的运行机制，详细说明了 DA_DIDS 的各个主要模块的功能，着重阐述了数据挖掘代理的工作流程，并对 DMA 所应用的数据挖掘的聚类分析算法，进行了详细的说明。首先介绍了聚类分析中的数据类型，然后说明了聚类分析中所遵守的准则，并详细介绍了三种聚类算法及如何应用它们进行聚类，和它们各自的优缺点。

第 5 章　DA_DIDS 系统性能测试及展望

5.1 聚类实验结果评价的标准

使用第 4 章的聚类算法对数据处理以后，需要评价聚类结果的好坏，尤其是对于多维特征向量的模式，不能直观的看出聚类效果，常用一下几个指标来考虑：

(1)聚类中心距离

下表 5-1 表明五个聚类中心相互之间的距离，从表中可以看出：Z_5 远离其他中心，而 Z_1 与 Z_2、Z_1 与 Z_4 是靠近的。

表 5-1 聚类中心距离

Table 5-1 clustering center distance

聚类中心	Z_2	Z_3	Z_4	Z_5
Z_1	3.7	12.6	1.9	52.1
Z_2		20.0	6.0	49.2
Z_3			15.1	35.6
Z_4				48.5

(2)聚类域中的样本数目

将聚类中心距离与相应的样本数目结合起来，可以更清楚地分析聚类结果是否正确，例如结合表 5-1，在 Z_5 的的聚类域中，如果只有少数几个样本，就可联系实际情况，分析 Z_5 是噪声还是确实是少数样本的聚类中心。

(3)聚类域中样本的距离方差

表 5-2 是 4 维属性样本的五个聚类域中的样本距离标准差。从列表的数值可以看出，S_1 聚类域内的样本的分布近似为超球体，S_5 域内样本则沿第 3 轴形成长条的超椭球体分布。

表 5-2 聚类域内样本距离标准差

Table 5-2　sample distance standard deviation in clustering

距离标准差 / 聚类域	σ_1	σ_2	σ_3	σ_4
S_1	1.1	0.8	0.8	1.3

S_2	2.3	1.2	1.6	0.8
S_3	3.4	4.6	7.1	11.5
S_4	0.4	1.0	0.9	1.3
S_5	3.9	4.9	17.6	2.8

此外，还可以利用其他距离度量值来分析模式样本的聚类性能，例如一种聚类域中离中心最远和最近的样本位置等。

5.2 聚类实验数据和结果

聚类分析算法使用的数据和下一章基于数据挖掘的入侵检测系统的实验平台使用的数据是 The UCI KDD Archive，Information and Computer Science，University of California, Irvine 模拟的入侵数据 Kdd99 Cup dataset[31, 32]，形式如下：

0, udp, private, SF, 105, 146, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0.00, 0.00, 0.00, 0.00, 1.00, 0.00, 0.00, 255, 254, 1.00, 0.01, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, normal.

0, udp, private, SF, 105, 147, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 0.00, 0.00, 0.00, 0.00, 1.00, 0.00, 0.00, 255, 254, 1.00, 0.01, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, snmpgetattack.

这些数据都是一条条的网络连接记录，第一个属性是连接时间，接着三个是分类属性：第一个表明 tcp 数据包还是 udp 数据包；第一个表示服务类型，如 http、tcp 等；第三个表明连接标记，如 SF、REJ、RSTR 等。接下来的是 37 个数值属性，是表明连接时的记录参数。最后一个属性则表明这条记录是正常的连接(normal)，还是入侵数据。实验时使用 360 个和 1000 个两种样本集。由于模拟数据中的入侵数据比较分散，所以将部分入侵数据记录挑出形成 360 个样本集；1000 个样本集数据是模拟真实入侵数据的一个片段。

k-均值算法和 k-中心点算法只能处理数值属性，而 k-原型聚类算法可以同时处理数值属性和分类属性[38, 45, 46]，但可以只用数值属性分析这些模拟数据，再从中提出分类属性，在随后介绍的关联规则使用，并可以和 k-原型聚类算法做比较。

由于划分的聚类分析算法都要预先输入聚类数目，本文首先使用 k-均值算法和 k-中心点算法对原始数据进行聚类观察算法分类不同的效果。

表 5-3 k-均值 5 类聚类结果

Table 5-3　5 kinds of clustering results of k-means

聚类数：5 迭代次数：9 迭代精度：0.000100 运行时间：0.10 秒				
聚类号	聚类数目	正常样本数目	百分比	异常样本数目
0	229	198	86.5%	31
1	35	35	100.0%	0
2	7	0	0.0%	5
3	5	5	100.0%	0

4	84	84	100.0%	0

表 5-4 k-均值 5 类聚类中心距离

Table 5-4 5 kinds of clustering center distancesof k-means

聚类中心	Z_0	Z_1	Z_2	Z_3
Z_1	338.69			
Z_2	7556.30	7556.30		
Z_3	1119.35	658.96	7720.56	
Z_4	120.27	251.53	7569.32	938.64

表 5-5 k-均值 10 类聚类结果

Table 5-5 10 kinds of clustering results of k-means

聚类数：10 迭代次数：32 迭代精度：0.000100 运行时间：0.27 秒				
聚类号	聚类数目	正常样本数目	百分比	异常样本数目
0	180	139	77.2%	41
1	16	16	100.0%	0
2	7	7	100.0%	0
3	6	6	100.0%	0
4	52	52	100.0%	0
5	4	4	100.0%	0
6	37	37	100.0%	0
7	5	0	0.0%	5
8	17	17	100.0%	0
9	56	56	100.0%	0

表 5-6 k-均值 10 类聚类中心距离

Table 5-6 10 kinds of clustering center distances of k-means

No	Z_0	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8
Z_1	356.36								
Z_2	310.78	539.15							
Z_3	1189.63	521.28	1067.87						
Z_4	68.57	376.69	323.68	947.43					

Z_5	1219.67	1307.65	1132.46	1738.65	1433.65				
Z_6	160.75	320.80	326.25	910.10	72.13	1229.35			
Z_7	7556.38	7670.69	7347.37	7765.56	7679.89	6220.63	7550.67		
Z_8	2700.26	186.26	465.31	778.41	310.16	1436.22	145.76	7550.65	
Z_9	40.10	390.34	290.24	1105.46	56.78	1190.56	123.85	7569.81	278.73

入侵数据聚类的目标是正常样本和入侵样本尽可能分开。对照表 5-3 和表 5-5 与对照表 5-4 和表 5-6 可以看出分为 10 类比 5 类效果要好些，因为对比同样的聚类域 Z_0，分为 10 类后所含的正常样本比较于分为 5 类的结果要少，它们都包含了 40 个入侵样本，这样对于设计入侵样本中心阈值(聚类域中正常样本所占百分比，小于该阈值的聚类中心认为是入侵样本中心)很有帮助。

从表 5-6 中可以看出，聚类域 Z_7 是远离其他类的，表 5-5 的统计结果也表明了这一点，这个域中的数据全是入侵数据，所以将此聚类域中心看做为入侵样本中心。同时考虑到聚类中心距离值很大，为了避免出现单个属性值影响聚类结果的出现，下一步本文进行属性值标准化实验，使用公式(4—5)。

表 5-7 k-均值属性标准化 10 类聚类结果

Table 5-7　10 kinds of standard clustering results of k-means

聚类数：10 迭代次数：5 迭代精度：0.000100 运行时间：0.04 秒				
聚类号	聚类数目	正常样本数目	百分比	异常样本数目
0	116	70	60.3%	46
1	21	21	100.0%	0
2	21	21	100.0%	0
3	11	11	100.0%	0
4	25	25	100.0%	0
5	6	6	100.0%	0
6	11	11	100.0%	0
7	47	46	97.9%	1
8	46	35	76.1%	11
9	56	56	100.0%	0

表 5-8 k-均值属性标准化 10 类聚类中心距离

Table 5-8　10 kinds of standard clustering center distance of k-means

No	Z_0	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8

Z_1	7.20								
Z_2	6.19	0.80							
Z_3	4.78	2.77	0.69						
Z_4	3.25	2.85	1.56	1.40					
Z_5	4.68	1.73	1.23	1.34	1.78				
Z_6	2.8	3.16	2.67	2.00	1.07	1.86			
Z_7	1.83	3.95	3.56	2.76	1.67	3.25	1.20		
Z_8	7.20	0.87	1.33	2.40	3.17	1.98	4.01	4.67	
Z_9	0.66	5.98	568	4.87	3.46	5.02	3.03	2.63	7.01

实验结果表 5-7 和表 5-8 可以看到，虽然标准化后距离值减小，但是聚类结果中正常样本和入侵样本的区别不如标准化之前，这表明聚类结果受单个属性值的影响很大，标准化消除了这种影响的同时也可能给聚类带来了副作用。下一步考虑的是特征选择和特征提取的问题，即哪些属性是主要的、有决定性作用的，哪些属性是次要可以合并的。

所谓特征选择，就是从 L 个度量值集合$\{x_1, x_2, \cdots, x_L\}$中，按某一准则选出聚类用的于集，作为降维(m 维，m < L)的分类特征；而特征提取，是使$(x_1, x_2, \cdots, x_L)$通过变换而产生 m 个特征$(y_1, y_2, \cdots, y_m)$。它们的目的，都是为了在尽可能保留原始信息的前提下，降低空间的维数，以达到有效的聚类。

本文试图使用聚类算法来分析特征(属性)的选择和提取问题。将 1000 个样本集中的数值属性取出，得到一个 1000×37 的矩阵，将其转置得到 37×1000 的矩阵作为聚类算法的处理对象，即输入为 37 条记录，每条记录有 1000 个属性值，对之进行聚类分析这些记录之间的相关性。

表 5-9 k-均值聚类 37 特征 10 类结果

Table 5-9 10 kinds of standard clustering center distance of k-means

聚类数：10 迭代次数：8 迭代精度：0.000100 运行时间：0.73 秒										
聚类号	0	1	2	3	4	5	6	7	8	9
聚类数目	2	1	1	0	1	1	2	2	2	25

从表 5-9 中可以看到 25 个属性都聚类在这个聚类域 Z_9 中，随后对这个域中的样本进行统计可以得到表 5-10 如下：

表 5-10 特征提取聚类域 Z_9 统计结果

Table 5-10　statistical results of character extraction from clustering field Z_9

序号	零的个数	最大值	最小值	备注(所有非零值)
3：	1000	0.0	0.0	
4：	1000	0.0	0.0	
5：	1000	0.0	0.0	
7：	998	4.0	0.0	奇异值：1.0 4.0
10：	998	1.0	0.0	奇异值：1.0 1.0
11：	1000	0.0	0.0	
13：	996	5.0	0.0	奇异值：3.0 1.0 1.0 5.0
14：	998	2.0	0.0	奇异值：2.0 1.0
15：	992	4.0	0.0	
16：	1000	0.0	0.0	
17：	997	1.0	0.0	奇异值：1.0 1.0 1.0
18：	997	1.0	0.0	奇异值：1.0 1.0 1.0
21：	989	1.0	0.0	
22：	990	1.0	0.0	
23：	996	1.0	0.0	奇异值：1.0 1.0 1.0 1.0
24：	996	1.0	0.0	奇异值：1.0 1.0 1.0 1.0
26：	996	1.0	0.0	奇异值：1.0 1.0 0.6 1.0
27：	525	1.0	0.0	
31：	880	1.0	0.0	
32：	746	1.0	0.0	
33：	866	1.0	0.0	
34：	989	0.8	0.0	
35：	990	1.0	0.0	
36：	952	1.0	0.0	
37：	969	1.0	0.0	

从统计结果可以看出，全为零的属性值可以删除，部分属性是可以合并的，此聚类域中大多数属性值都是零，只有个别是非零值，记录个别的奇异值，一是在判断是否入侵检测数据时，只要判断该属性值是否奇异值就行了；二是在关联规则特征提取之后生成入侵模式库时可以使用到。

有了以上的结果后，下面的实验是去除了部分属性的进行的 k-中心点算法对 360 个样本点数据进行聚类的结果。

表 5-11 k-中心点 10 类聚类结果

Table 5-11　10 kinds of clustering results of k-medoids

聚类数：10 迭代次数：359 运行时间：8.36 秒				
聚类中心序号：0 321 124 323 96 37 356 62 99 213				
聚类号	聚类数目	正常样本数目	百分比	异常样本数

				目
0	69	27	39.1%	42
1	7	7	100.0%	0
2	2	2	100.0%	0
3	3	3	100.0%	0
4	78	78	100.0%	0
5	139	130	93.5%	9
6	36	36	100.0%	0
7	3	0	0.0%	5
8	20	20	100.0%	0
9	3	3	100.0%	0

表 5-12 k-中心点 10 类聚类中心距离

Table 5-12 10 kinds of clustering center distances of k-medoids

No	Z_0	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8
Z_1	569.13								
Z_2	1710.37	1804.78							
Z_3	1697.39	1215.23	2400.43						
Z_4	87.68	496.69	1650.35	1479.98					
Z_5	25.66	550.54	1701.56	1680.56	60.32				
Z_6	201.78	359.55	1656.89	1541.03	134.35	186.31			
Z_7	7558.64	7756.97	6180.39	7798.48	7658.69	7710.68	7698.63		
Z_8	380.28	186.09	1598.69	1400.36	312.21	363.79	168.39	7556.49	
Z_9	321.49	624.21	1321.46	1805.52	320.46	298.89	356.67	7426.35	461.91

对照表 5-11 和表 5-5，可以发现 k-中心点算法比 k-均值算法的聚类效果有所提高，因为相同入侵样本聚类域中的正常样本数量减少了很多，但是运行时间和迭代次数明显增多。随后进行 1000 个样本点实验，k-中心点算法已经不能胜任， 而 k-均值算法在增到 4000 个样本点以上时也不能胜任。由于 k-中心点算法的运算复杂度太高，通常针对超大数据集的 k-中心点算法改进都是采取局部采样的方法降低复杂度，如 CLARA 和 CLARANS 算法以，但是算法准确度降低了，而且可能漏掉入侵数据样本，这对最终的聚类结果是有很大影响的。本文采取的是对 k-均值算法加以改进，进行超大入侵数据集的批量处理，主要有以

下几点：

1、将入侵样本分割成 n 个小的于样本集，每个大小不超过 2000；

2、对第一个于样本集进行 k-均值算法聚类，得到 k 个聚类中心；将这 k 个聚类中心作为 k 个样本加入到第二个子样本集，并作为初试聚类中心，进行 k-均值聚类……重复进行直到第 n 个子样本集聚类完成；

3、统计各个样本集聚类结果，最终得到 k 个聚类中心。

以下是对 20000 个入侵记录分割成 10 个子集进行 k-均值算法的结果。

表 5-13 改进 k-均值 10 类聚类第 10 个子集聚类结果

Table 5-13 the 10th subclustering results of improved k-means

聚类数：10 迭代次数：16 迭代精度：0.000100 运行时间：6.67 秒				
聚类号	聚类数目	正常样本数目	百分比	异常样本数目
0	5	4	80.0%	1
1	89	1	1.1%	88
2	3	3	100.0%	0
3	2	2	100.0%	0
4	249	242	97.2%	7
5	440	437	99.3%	3
6	33	33	100.0%	0
7	988	771	78.0%	227
8	189	188	99.5%	1
9	3	3	100.0%	0

表 5-14 改进 k-均值 10 类聚类中心距离

Table 5-14 10 kinds of clustering center distance of improved k-means

No	Z_0	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8
Z_1	66.35								
Z_2	13010.66	13021.58							
Z_3	160.38	180.46	12797.32						
Z_4	12.76	78.43	12797.35	161.34					
Z_5	12.74	72.34	12797.38	159.93	0.83				
Z_6	14.95	77.64	12797.333	149.74	11.37	11.65			
Z_7	12.86	76.58	12797.35	159.89	1.42	0.56	11.95		
Z_8	12.72	76.78	12797.36	156.71	2.02	2.91	8.73	3.24	

Z_9	321.27	263.35	12530.25	357.31	341.23	347.76	342.75	344.35	342.75

从表中可以看出聚类域 Z_1 和 Z_7 入侵样本比较多，而对比前几个子集的聚类结果可以发现相同的结果，说明前一个聚类结果对后一个子集聚类是有很大影响的，如果在第一个子集聚类的时候就能较好的把正常样本和入侵样本分开，在随后的聚类中正常和入侵样本的检测就会向各自的聚类中心更加趋近。实验结果表明改进的 k-均值算法对于大数据集是可行的。

k-均值算法和 k-中心点算法处理的都只是数值属性，k-原型算法可以处理分类和数值属性，分类属性对整个结果的影响由算法中的权值系数 γ 决定。以下是 k-原型算法运行结果：

表 5-15 k-原型 10 类聚类中心距离

Table 5-15 10 kinds of clustering center distance of improved k-prototypes

聚类数：10 迭代次数：11 运行时间：13.96 秒 r=40				
聚类号	聚类数目	正常样本数目	百分比	异常样本数目
1	192	185	96.4%	7
2	8	2	25.0%	6
3	24	24	100.0%	0
4	16	0	0.0%	16
5	11	0	0.0%	11
6	58	58	100.0%	0
7	3	3	100.0%	0
8	5	0	0.0%	5
9	28	28	100.0%	0
10	15	4	26.7	11

表 5-16 k-原型 10 类聚类中心距离

Table 5-16 10 kinds of clustering center distance of improved k-means

No	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8	Z_9
Z_0	0.350								
Z_1	0.178	0.355							
Z_2	0.525	0.530	0.535						
Z_3	1.260	1.230	1.162	1.118					

Z_4	0.142	0.361	0.228	0.568	1.172				
Z_5	2.186	2.184	2.171	2.212	2.450	2.210			
Z_6	1.839	1.871	1.875	1.201	2.020	1.856	2.851		
Z_7	0.210	0.275	0.251	0.520	1.136	0.245	2.168	1.856	
Z_8	0.512	0.521	0.496	0.713	1.201	0.561	2.226	1.876	0.532

从表中可以看出，聚类域Z_2、Z_4、Z_5和Z_8的聚类效果都相当不错，入侵样本和正常样本可以很好的区分开来，这样的聚类中心就可以作为入侵样本中心。但是 k-原型算法运算时需要统计出全部的分类属性值再加以数值化，这就意味着需要一次扫描所有的样本的分类属性值加上权值系数γ后再作为数值属性参与到计算中，增大了运行时间和运算复杂度。

5.3 关联规则

关联规则[34]挖掘大量数据中项集之间有趣的关联或相关联系。随着大量数据不停的收集和存储，许多业界人士对于在他们的数据库中挖掘关联规则越来越感兴趣。从大量商务记录中发现有趣的关联关系，可以帮助许多商务决策的制定，如分类设计、交叉购物和贱卖分析。关联规则在决策支持系统、专家系统和智能系统中得到广泛的应用，尤其在近几年倍受人们关注。例如，在学生成绩数据库中，可以发现“离散数学成绩为优的人编译原理成绩也是优的可能性是 66%”；在超市的购买记录中，可以发现“同时购买奶粉和咖啡的顾客又购买了方糖的可能性是 86%”。发现这些规则，我们可以加强离散数学课的教学，以此来提高编译原理课的教学效果；或者降低奶粉和咖啡的价格来促进方糖的销售。

如果不考虑关联规则的支持度和可信度，那么在事务数据库中存在无穷多的关联规则。事实上，人们一般只对满足一定的支持度和可信度的关联规则感兴趣。因此，为了发现出有意义的关联规则，需要给定两个阈值：最小支持度和最小可信度。前者即用户规定的关联规则必须满足的最小支持度，它表示了一组物品集在统计意义上的需满足的最低程度；后者即用户规定的关联规则必须满足的最小可信度，它反应了关联规则的最低可靠度。

在实际情况下，一种更有用的关联规则是泛化关联规则。因为物品概念间存在一种层次关系，如夹克衫、滑雪衫属于外套类，外套、衬衣又属于衣服类。有了层次关系后，可以帮助发现一些更多的有意义的规则。例如，“买外套⇒买鞋子”（此处，外套和鞋子是较高层次上的物品或概念，因而该规则是一种泛化的关联规则）。由于商店或超市中有成千上万种物品，平均来讲，每种物品（如滑雪衫）的支持度很低，因此有时难以发现有用规则；但如果考虑到较高层次的物品（如外套），则其支持度就较高，从而可能发现有用的规则。

另外，关联规则发现的思路还可以用于序列模式发现。用户在购买物品时，除了具有上述关联规律；还有时间上或序列上的规律，因为，很多时候顾客会这次买这些东西，下次买同上次有关的一些东西，接着又买有关的某些东西。

5.3.1 基本概念

设$I=\{i_1,i_2,\cdots,i_m\}$是所有项的集合。D 是事务集合，每个事务 T 是项的集合，且 T⊆I。每一个事务 T 都有一个唯一的标识符 TID。设有项集$A=\{p_1\wedge p_2\wedge\cdots\wedge p_n\}$和$B=\{q_1\wedge q_2\wedge\cdots\wedge q_m\}$，且$A\subset I,B\subset I,A\cap B=\phi$。关联规则(Association Rule)是形如 A⇒B 的蕴涵式。规则 A⇒B 在事务集 D 中成立，具有支持度 S(Support)，其中 S 是 D 中事务包含$A\cup B$的概率 P($A\cup B$)。规则 A⇒B 在事务集 D 中成立，具有可信度 c(confidence)，其中 c 是 D 中事务包含 A 的同时也包含 B 的概率 P(B|A)，即

$$Support(A\Rightarrow B)=p(A\cup B) \tag{4-22}$$

同时满足最小支持度阈值(min_sup)和最小可信度阈值(min_conf)的规则称作强规则。通常使用 0%和 100%之间的值而不是用 0 和 1 来表示支持度和可信度。包含 k 个项集的称为 k-项集。如果项集满足最小支持度，则称它为频繁项集(frequent itemset)。频繁 k-项集的集合通常记为L_k。

关联规则的挖掘主要分两个步骤：

1)找出所有的频繁项集，即项集出现的频繁性不小于预定义的最小支持度计数。

2) 由频繁项集产生强关联规则，即同时满足最小支持度和最小可信度的规则。

这两步中，第二步相对容易。关联规则算法的性能由第一步决定。

5.3.2 Apriori 算法

Apriori 算法是挖掘关联规则频繁项集中的一种基本算法。Apriori 算法使用频繁项集性质的先验知识。该算法使用一种称为逐层搜索的迭代方法，k-项集用来搜索(k+1)-项集。首先，找出频繁 1-项集的集合，记做 L_1。L_1用来找频繁 2-项集的集合L_2，L_2用来找L_3，如此下去，直到不能找到频繁 k-项集为止。

为了提高频繁项集逐层产生的效率，一种称为 Apriori 性质的重要性质用来压缩搜索空间：频繁项集的所有非空子集必须也是频繁的。Apriori 基于以下事实：根据定义，如果项集 I 不满足最小支持度 min_sup，则项集 I 不是频繁的(P(I)

<min_sup)，此时如果把项集 A 添加到 I，则结果项集 I∪A 不可能比 I 更频繁出现，因此，I∪A 也不是频繁的(P(I∪A)<min_sup)。该性质属于一种特殊的分类，称为反单调(anti-monotone)，指一个集合不能通过测试，则它的所有超集也不能通过相同的测试。称之为反单调是因为在通不过测试的意义下，读性质是单调的。

Apriori 算法中寻找频繁项集分为连接和剪枝两步[14]：

1)连接：为找L_k，通过L_{k-1}与自身连接产生候选 k-项集的集合，该候选集记做C_k。设l_1和l_2是L_{k-1}中的项集。记号l_1[j]的l_1的第 j 项(例如，l_1[k−2]表示l_1的第 k−2 项)。为方便计，假定事务和项集中的项按字典次序排序。当L_{k-1}的前 k−2 项相同时，L_{k-1}的项是可连接的。此时称他是可连接的，即如果(l_1[1]= l_2[1])∧(l_1[2]= l_2[2])∧…∧(l_1[k-2]= l_2[k-2])∧(l_1[k-1]= l_2[k-1])成立，可执行连接L_{k-1}。此时连接l_1和l_2可得到项集l_1[1] l_1[2]…l_1[k-1] l_2[k-1]。

2)剪枝：C_k是L_k自身连接产生的超集，它的子集可能是频繁的也可能是不频繁的，但是所有的频繁 k-项集都包含在C_k中。扫描事务集确定C_k中每个候选集的计数从而确定L_k(计数值不小于最小支持度计数是频繁的且属于L_k)。

为了减少产生C_k的计算量，可以使用 Apriori 性质：任何非频繁的(k-1)-项集都不可能是频繁 k-项集的子集。因此，如果一个候选 k-项集的(k−1)-子集不在人L_{k-1}中，则该候选集也不可能是频繁的，从而可以从C_k中删除。通过保存所有频繁于集的哈希树可以快速的完成这种于集测试。

5.3.3 关联规则实验数据和结果

应用关联规则的目的是对入侵检测数据的分类属性进行相关分析建立入侵检测模型。因此算法只须处理入侵数据分类属性而不用考虑数值属性。采用的样本集依旧是 360 和 1000 两类，从中提取每个样本的分类属性，形式如表 5-17。

表 5-17 入侵数据集的分类属性值

Table 5-17 classified attribute value of intrusion data set

序号	TCP/UDP	协议	标记	类型
1	tcp	other	REJ	Httptunnel
2	tcp	http	SF	Normal

3	tcp	other	REJ	Httptunnel
4	tcp	pop3	SF	Guess_passwd
5	tcp	Smtp	SF	Normal

由于 Apriori 算法的复杂度很高，对于 n 项集，在 n 比较小的情况下算法实现很理想；可是当 n 比较大时，该算法在第一次连接的时候就要产生 n(n—1)/2 个候选 2-项集，接着进行大量的剪枝操作，内存开销很大。对此，有两点改进可以用来减小算法复杂度：

1、连接和剪枝同时进行：每产生一(k+1)候选集，就对该候选集进行支持度计数，如果不满足最小的支持度，则可以删除该项集。

2、剪枝时，对于所有生成的项集有序排列：在使用 k-项集生成(k+1)-项集时，保存上一步生成的频繁项集，查找去除(k+1)-候选集中的一个元素后得到的子集在 k-项集中是否存在，根据 Apriori 性质，如果不存在该子集，则表明该(k+1)-项集不频繁。

实验步骤：

第一步：进行类型统计生成项集，包括协议类型，状态类型和攻击类型；

表 5-18 入侵数据集的分类属性统计结果

Table 5-18 classified attribute statistic result of intrusion data set

协议类型	总计	标记类型	总计	状态类型	总计
other	21	REJ	17	httptunnel	21
http	267	SF	330	normal	306
smtp	30	RSTR	8	portsweep	9
pop3	6	SH	3	warezmaster	16
private	12	81	1	guess_passwd	5
ftp	5	OTH	1	nmap	3
ftp_data	16				
finger	3				

第二步：使用 Apiori 生成频繁项集(最小支持度为 2)；

表 5-19 Apiori 算法生成的频繁项集

Table 5-19 frequent item set created by Apiori algorithm

运行时间：8.02 秒	
频繁项集	支持度计数
Other; REJ; httptunnel;	17
Other; SF; httptunnel;	4
http; SF; normal;	266
Smtp; SF; normal;	30
pop3; SF; guess_passwd;	5
Private;RSTR;portsweep;	8
ftp; SF; warezmaster;	5
ftp_data; SF; normal;	5

ftp_data; SF; warezmaster;	11
Finger; SF; normal;	3

第三步：使用频繁项集生成关联规则。因为我们只关心入侵数据状态类型，所以不必生成所有的关联规则，只需生成能够推导出状态类型的关联规则，如表 5-20 所示。

表 5-20 由表 5-19 生成的关联规则

Table 5-20 association rules created from table 5-19

关联规则(A⇒B)	项集 A 与 B 计数	项集 A 计数	支持度	可信度
other∧REJ⇒httptunnel	17	17	4.62%	100.00%
other∧SF⇒httptunnel	4	4	1.21%	100.00%
http∧SF⇒normal	266	266	74.89%	100.00%
smtp∧SF⇒normal	30	30	8.32%	100.00%
pop3∧SF⇒guess_passwd	5	6	1.23%	82.33%
private∧RSTR⇒portsweep	8	8	2.32%	100.00%
ftp∧SF⇒warezmaster	5	5	1.23%	100.00%
ftp_data∧SF⇒normal	5	16	1.23%	31.25%
ftp_data∧SF⇒warezmaster	11	16	2.98%	68.75%
finger∧SF⇒normal	3	3	0.86%	100.00%

对于可信度为 100%的强关联规则就可以提出来加入到入侵模型库中，如上表中的第一条规则表示协议类型为 other，标记为 REJ 的记录肯定是攻击类型为 httptunnel 的入侵数据，这样的记录存在的可能性是 4．72%(支持度)。

对于大数据集例如 1000 个样本点，从其实验结果中可以看到，Apiori 算法的运行时间明显加长，如表 5-21 所示。

表 5-21 Apiori 算法 1000 个样本集得到的关联规则

Table 5-21 association rules created from 1000 samples using Apiori algorithm

关联规则(A⇒B)	项集 A 与 B 计数	项集 A 计数	支持度	可信度
http∧SF⇒normal	816	816	81.60%	100.00%
smtp∧SF⇒normal	109	110	10.90%	99.09%
ftp_data∧SF⇒normal	27	30	2.70%	90.00%
ftp_data∧SF⇒named	2	30	0.20	6.67%
ftp∧SF⇒normal	3	5	0.30%	60.00%
other∧SF⇒named	2	3	0.20%	66.67%
auth∧SF⇒normal	7	8	0.70%	87.50%
X11∧SF⇒named	2	5	0.20%	40.00%
X11∧SF⇒xlock	3	5	0.30%	60.00%
finger∧SF⇒normal	2	2	0.20%	100.00%

time∧SF⇒normal	2	2	0.20%	100.00%
telnet∧SF⇒normal	2	8	0.20%	25.00%
telnet∧SF⇒xlock	2	8	0.20%	25.00%
telnet∧SF⇒multihop	2	8	0.20%	25.00%

注：运行时间 32.03 秒

5.4 DA_DIDS 的应用前景

在互联网高速上网迅速发展的今天，随着安全事件的急剧增加以及入侵检测技术逐步成熟，入侵检测系统将会有很大的应用前景。比如银行的互联网应用系统(支付网关、网上银行等)，科研单位的开发系统，军事系统，普通的电子商务系统，ICP 等，都需要有 DA_DIDS 的护卫。

(1)无线网络。移动通信由于具有不受地理位置的限制、可自由移动等优点而得到了用户的普遍欢迎和广泛使用。但是移动通信在带来可移动的优越性的同时也带来系统安全性的问题。由于移动通信的固有特点，移动台(MS)与基站(BS)之间的空中无线接口是开放的，这样整个通信过程，包括通信、链路的建立、信息的传输(如用户的身份信息、位置信息、语音以及其它数据流)均暴露在第三方面前；而且在移动通信系统中，移动用户与网络之间不存在固定物理连接的特点使得移动用户必须通过无线信道传递其身份信息，以便于网络端能正确鉴别移动用户的身份，而这些信息就可能被第三者截获，并伪造信息，假冒此用户身份使用通信服务；另外无线网络也容易受到黑客和病毒的攻击。因此，DA_DIDS 在无线网络方面有广阔的应用前景。

(2)DA_DIDS 走进家庭。最近几年来 IDS 让公司经理们不必担心黑客和内部作案人员，现在 IDS 要保护家庭里的个人计算机。越来越多的人通过互联网将家中的计算机与公司联网。而由于使用了宽带，这些用户的网络或 DSL 调制解调器总是处于打开的状态，对黑客有极大的诱惑力。黑客可以由此侵入网络，盗窃信用卡、身份证明或进入网络管理系统，一些传播很快的病毒还会把个人计算机的内容暴露给黑客，这些都预示着 IDS 将逐渐走入家庭。DA_DIDS 作为 IDS 的发展方向必将得到人们的认可。

5.5 小结

本章介绍了数据挖掘聚类算法的实验结果评价标准，并用关联规则对数据进行了实验。我们在 Linux 环境下编程，用 MYSQL 建立了一个基于数据挖掘的入侵检测系统模型。模型使用了 40000 个训练样本点得到 10 个聚类中心。再使用 Apriori 算法计算全部训练样本中的 24180 个入侵样本点的分类属性得到 737 条关联规则，其中 627 条强关联规则作为入侵模型，运行时间为 690.63 秒。随后输入 20000 个样本点进行入侵判断实验，属于入侵聚类域的新增样本有 6543 个，其中 96.58%是入侵样本，说明入侵样本中心的选择是正确的，聚类效果也比较好。入侵检测结果在 7554 个入侵样本中发现入侵样本数为 6319 个，发现的可疑入侵样本个数为 183，漏报率为 13.93%，误报率为 1.04%，系统可靠性(正确识别入侵样本的百分比)为 80.88%，这表明该系统检测是比较成功的。系统运行时间 51.98 秒，比较短，可以满足入侵检测系统的实时性要求。实验结果表明，主要的性能参数：有效性、强度、效率、资源消耗率和抗攻击能力都完全达到了

我们的设计目标，该基于数据挖掘的分布式入侵检测系统模型对网络入侵的检测是成功的。

结论

数据挖掘是入侵检测的一个重要应用方向，而基于数据挖掘技术的入侵检测研究在国内尚处于初级阶段，本文的工作为这方面的进一步研究提供了一些可行的分析工具和方法，也为今后进一步开发相应的入侵检测产品提供了实现的基本手段。

本文介绍了数据挖掘在入侵检测中的应用，并在此基础上提出了一种基于数据挖掘和 Agent 的入侵检测系统。它可以有效地解决现有的许多入侵检测系统中的问题，构造高效灵活而且具有良好扩展性的入侵检测系统。现有的大多数入侵检测系统具有自适应性差、误报警率和漏报警率高以及需要承载大量数据等缺陷，而应用数据挖掘技术的入侵检测系统较好地解决了上述问题，它应用数据挖掘技术自主地从网络数据中获取有用的知识，DA_DIDS 系统采用的特征分类建立规则库和数据挖掘技术可以有效地对系统的规则库进行更新，保持着最新和最有效率的规则库，很大程度上提高了入侵检测的效率和准确性。

在研究和实现过程中，我们发现 DA_DIDS 还存在一些缺陷，这也是将来进一步需要研究的工作：

(1) 进一步提高数据挖掘算法的效率。

(2) 加强与其它防护系统协同技术的研究。

(3) 进一步完善安全通信机制。

(4) 进行更广泛的性能测试。

最后我们应该明白的是：IDS 尽管是计算机网络安全的重要组成部分，但它不是一个完全的计算机网络系统安全解决方案，它不能替代其他安全系统如：访问控制、身份识别与认证、加密、防火墙、病毒的检测与杀除等的功能。但可以将它与其他安全系统，如防火墙系统、安全网管系统等增强协作，以增加其自身的动态灵活反应及免疫能力，为我们提供更加安全的网络环境。

参考文献

[1]. Lee W, Stolfo S J. Data mining approaches for intrusion detection. Proceedings of the 7th USENEX Security Symposium, January, 1998.

[2]. Lee. W., Stolfo, S. J., Mot, K. Datamining in work flow environments: Experience in intrusion detection [A]. In:Chaudhuri, S ed. Proceedings of the 1999 Conference on Knowledge Discovery and Data Mining[C]. CA:ACM Press, 1999.

[3]. Eskin E. Signal Detection over Noisy Data Using Lzamed Probability Distributions(C 1. Proceeding of the Seventeenth) International Confere on Machine Learning(ICML-2000),2000.

[4]. Richard P. Lippmann, David J. Fried, Isaac Graf etc. Evaluating intrusion Detection Systems:The 1998 DARPA Off-line Intrusion Detection Evaluation[R]. In Proceedings of the 2000 DARPA Information Survivability Conference and Exposition. 2000.

[5]. Krishna, K. Murty, M.N.Genetic K-Means Algorithm. IEEE Transactions on Systems, Man and Cybernetics (PartB), 1999, 29(3):433-439.

[6]. Hu Yingwei, Jiang Zhongxiang. The Techniques and Development of Network Security.Proceedings of the 1998 2nd International Conference on Information Infrastructure. Beijing China. 1998: Publishing House of Electronics Industry: 541-545.

[7]. T. Fawcett and F. Provost. Combining data mining and machine learning for effective user profiling[A]. In: Proceedings of the 2nd International Conference on Knowledge Discovery andData Mining[C], Portland, OR, AAAI Press. 1996, 8-13.

[8]. J. P. Anderson. Computer Security Threat Monitoring and Surveillance. Fort Washington, Pennsylvania. April 1980 [5]. CIDF specification document. Communication in the Common Intrusion Detection Framework v0.7.DRAFT Specification 8 June 1998

[9]. Eric Cole 著，苏雷 译，黑客攻击透析和防范[M]北京:电子工业出版社,2002

[10]. Bruce Schneie 著， 吴世忠，马芳 译， 网络信息安全的真相[M]北京:机械工业出版社，2001.

[11]. CIDF specification document. Communication in the Common Intrusion Detection Framework v0.7. DRAFT Specification 8 June 1998

[12]. Lee,W .A Data Mining framework for Constructing Features and Models for

Intrusion Detection Systems[PhD Thesis].Columbia University,USA, 1999.

[13]. Coit C J, Staniford S, McAlerney J. Towards faster patern matching for intrusion detection or exceeding the speed of snort[C]. In: DARPA Information Survivability Conference and Exposition, 2001

[14]. S Staniford Chen, Cheung S. GrIDS-A graph based intrusion detection system for large networks. In: The 19th National Information Systems Security Conference. Baltimore, Maryland, USA, 1996. 361ь370. http: //seclab. cs. Ucdavis. edu/papers/nissc96. pdf

[15]. Fayyad U M, Piatets Shapiro G, Smyth P. Advance in knowledge discover and data mining[M]. Califoiriia:AAAI Pre99, The MIT Press 1996:1-25

[16]. Han J, Kamber Morgan Kaufmann M. Data Mining:Concepts and Techniques 2000 [M]. New York:Series Editor Morgan Kaufmann Publishers, 2000. 1-200.

[17]. Cecilia M Procopiuc. Clustering problems and their applications (survey) [DB/OL]. Department of Computer Science, Duke University 1997. http://www. cs. duke. edu/magda

[18]. Application of Data Mining to Intrusion Detection. http://www. isse. gmu. edu/csis/infs765/handouts/handout12. pdf, 2000

[19]. Wenke Lee. A Data Mining Framework for Constructing Features and Models for Intrusion Detection [EB/OL], http://www. csc. ncsu. edu/faculty/lee/papers/thesis. ps. gz.

[20]. Wenke Lee, Salvatore J. Stolfo, Kui W. Mok. A Data Mining Framework for Adaptive Intrusion Detection [EB/OL], http://www. cs. columbia. edu/sal/hpapers/framework. ps. gz

[21]. Dorothy E. Denning. An Intrusion-Detection Model. IEEE Transactions on software engineering [J], VOL. SE-13, NO. 2 , February 1987 , 222-232

[22]. Wayne Jansen, Peter Mell, Tom Karygiannis, at al. Applying Mobile Agents to Intrusion and Response[Z]. NIST Report (IR)-416, 1999, (10).

[23]. 刘海峰, 卿斯汉, 等。一种基于审计的入侵检测模型及其实现机制[J]电子学报, 2002, 30(8):1167-1171

[24]. Wenke Lee, Salvatore J. Real Time Data Mining-based Intrusion Detection [EB/OL], http://www. cs. columbia. edu/ids/publications/ dmids-discex01. ps.

[25]. Fisher, D. H. Knowledge acquisition via incremental conceptual clusterings [J]. Maching Learning 1995, 2:139-172

[26]. W. Lee, S. J. Stolfo, and K. W. Mok. Mining in a data-flow environment: Experience in network intrusion detection [A]. In:Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery&Data Mining (KDD-99) [C], 1999.

[27]. W. Lee, S. J. Stolfo, and K. W. Mok. Mining audit data to build intrusion detection models [A]. In:Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining[C], New York, NY, AAAI Press. 1998.

[28]. M. Crosbie and E. H. Spafford. Active defense of a computer system using autonomous agents[R]. Technical Report CSD-TR-95-008, COAST Laboratory, Department of Computer Science, Purdue University, West Lafayette, IN, 1995.

[29]. W. Lee and S. J. Stolfo. Data mining approaches for intrusion detection [A]. In: Proceedings of the 7th USENIX Security Symposium[C], San Antonio, TX, 1998.

[30]. W. Lee, S. J. Stolfo, and K. W. Mok. A data mining framework for building intrusion detection models[A]. In: Proceedings of the 1999 IEEE Symposium on Security and Privacy[C], Rerkelv. California. 1999. 120-132.

[31]. Kdd99 Cup dataset. http://kdd. ics. uci. edu/databases/kddcup99/kddcup99. html, 1999

[32]. Barbara D. ADAM: Detecting Intrusions by Data Mining. Proceedings of IEEE Workshop on Information Assurance and Security, 2001

[33]. Helmer G G, Wong J S K, Honavar V, et al. Intelligent Agents for Intrusion Detection. In: Proceedings of IEEE Information Technology Conference, Syracuse, NY, 1998-09:121-124

[34]. Agrawal R, Strikarit R. Fast algorithms for mining association rules in large databases[C]. In:Proceedings of VLDB, Santiago Chile, 1994:487-499

[35]. Jeremy Frank. Artificial Intelligence and Intrusion Detection:Current and Future Direction [C]Proceeding of 17th National Computer Security Conference. http://www. nasa. gov/ ic/people/frank, 2001

[36]. Wenke Lee, Salvatore J. Stolfo, Kui W. Mok, Mining Audit Data to Build Intrusion Detection Models, http://www. csee. umbc. edu/cadip/does/Net world intrusion/kdd98. ps, 1998

[37]. Wenke Lee, Salvatore J. Stolfo, Kui W. Mok Adaptive Intrusion Detection: a Data Mining Approach[J], Articial Intelligence Review, 2001, 14:533-567

[38]. W Fan. Cost-senstive, Scalable and Adaptive Learning Using Ensemble-based Methods[D]. PhD thesis, Columbia University, 2001

[39]. Mukkamala R, Gagnon J, Jajodia S . Integrating data mining techniques with

intrusion detection methods[A]. In:Atluri and John Hale, Research Advances in Database and Information Systems Security[C]. Boston, MA:Kluwer Publishers, 2000. 33-46

[40]. Eskin E, Arnold A, Prerau M, et al. A geometric framework for unsupervised anomaly detection:detecting intrusions in unlabeled data[A] Applications of Data Mining in Computer Security[C] Kluwer Academi Publisher, Boston , 2002. 77-102

[41]. Honig A, Howard A, Eskin E, et al. Adaptive model generation: an architecture for the deployment of data mining-based intrusion detection syste ms[A]. Applications of Data Mining in Computer Security[C] Kluwer Academic Publisher, Boston, 2002. 153-194

[42]. Schultz M, Eskin E, Zadok E, et al. Data mining methods for detection new malicious executables [A]. In Proceedings of IEEE Symposium on Security and Privacy(IEEE S&P-2001)[C]Oakland, CA, May 2001. 38-49

[43]. Eskin E. Anomaly detection over noisy data using learned probability distributions[A] .In Proceeding International Conference on Machine Learning[C]Morgan Kaufmann Press, Palo Alto, CA, 2000. 255-262

[44]. DARPA. CIDF-Common Intrusion Detection Framework [DB/01]. http://www.gi.dos.org/drafts/architecture.txt, 2001-07-10.

[45]. Maulik,U., Bandyopadhyay, S. Genetic algorithm-based clustering technique. Pattern Recognition, 2000,3 3(9):1455-1465.

[46]. Portnoy, L., Eskin, E., Stolfo, S.J. Intrusion detection with unlabeled data using clustering. In:Barbara, D ed. Proceedings of ACM CSS Workshop on Datamining Applied to Security. Philadelphia:ACM Press, 2001.

致谢

在这论文完成之际，衷心感谢所有关心和帮助过我的人们。

首先感谢我的导师贾瑞新教授，在课题研究和论文撰写的每个阶段，贾老师都提出了许多建设性的建议，在我论文的每一部分，都凝结了贾老师的心血。贾老师具有渊博的学识和长者风度，对工作兢兢业业，做学问一丝不苟，学习和生活上都给于了我极大的帮助，贾老师对待工作对待人生的态度将是我一生的榜样。

感谢我最爱的人，正是她的不断的关心和支持，给了我源源不断的动力，使我不断战胜自我，超越自我。

感谢秦华老师，蒋丹老师在学习和研究工作给予我的无私帮助。

感谢我的朝夕相处的室友以及师兄师弟师妹们，我们如兄弟姐妹般的相处，对 CS、对社会、对人生的诸多讨论都会给我留下深刻回忆。

再次衷心感谢。