应用入侵检测研究与进展*)

林冬梅[1] 钟 勇[1,2] 秦小麟[2]
(佛山科学技术学院信息与教育技术中心 佛山 528000)[1]
(南京航空航天大学计算机科学与技术系 南京 210016)[2]

摘 要 传统的基于网络和主机入侵检测系统有不足,促使应用入侵检测的研究得到重视。本文总结了应用入侵检测技术的研究现状及其进展,指出应用入侵检测技术研究的难点和需要解决的关键问题,对今后的发展趋势进行了分析和展望。

关键词 应用入侵检测,入侵检测,信息安全

Research and Development of Application Intrusion Detection

LIN Dong-Mei[1] ZHONG Yong[1,2] QIN Xiao-Lin[2]
(Information and Educational Technology Center, Foshan University, Foshan 528000)[1]
(Information Science and Technology Institute, Nanjing University of Aeronautics and Astronautics, Nanjing 210016)[2]

Abstract *Traditional network-based intrusion detection system and host-based intrusion detection system are not enough, which make researches of application intrusion detection get attentions. The paper summaries the status and development of application intrusion detection, and points out the difficulties of research and the existing problems that need be solved. Finally, then future direction in this field is analyzed and discussed.*

Keywords Application intrusion detection, Intrusion detection, Information security

1 引言

随着信息技术在社会各个领域的深入开展,人们在得益于信息革命所带来的巨大机遇的同时,也面临着信息安全问题的严峻考验。由于信息系统安全脆弱性的客观存在,操作系统、应用软件、硬件设备不可避免地会存在一些安全漏洞。而经验也显示不可能百分之百地预防所有的安全问题。黑客们常常使人吃惊,因为他们总能发现新的方法闯入或干涉我们的系统[1]。

随着入侵手段日益的多元化、复杂化、智能化,由于不能主动跟踪入侵者和预先防范入侵,传统的被动型安全机制,如以身份认证为基础的存取控制技术、控制信息流向的多级安全机制、数据加密和防火墙等技术,难以胜任安全形势的需要。入侵检测系统(Intrusion Detective System, IDS)作为一种主动的安全防范机制,是安全检测的最后一道防线,受到越来越广泛的重视。

在多年研究中,入侵检测技术已取得长足的进步[2,3]。但这些研究大多集中在基于网络(Network-based)或主机(Host-based)的入侵检测系统上[4,5]。

网络入侵检测系统(Network-Based Intrusion Detection System, NIDS)用于审计用户的网络活动,如分析网络流量、协议、网络管理等数据来检测入侵,典型的系统如 Snort、NFR、Bro 等。虽然当前 NIDS 系统的研究已较为完善,但还存在一些难以解决的问题[6]:(1) 大多数 NIDS 依靠对网络信息的侦听,对加密的网络信息则难以或无法侦听;(2) 许多 Internet 标准,如 RFC 只定义了正确协议语义,没有描述应用系统对偏差数据如何反应,造成不同的应用或不同的应用版本对相同偏差数据有不同的反应,而 NIDS 无法完全了解各类应用的不同反应,从而达不到相应的检测精度;(3) 网络带宽的不断提高也使 NIDS 难以达到线速或提高交换带率,从而影响 NIDS 的部署和使用。

主机入侵检测系统(Host-based Detection System, HIDS)用于审计主机用户在操作系统级的活动,如用户的主机登录、操作系统命令、资源使用等,典型的 HIDS 系统如 Haystack[7], Denning 的入侵检测模型[8]等。多数 HIDS 系统往往通过主机日志来获取异常检测数据,而这时入侵可能已经发生,HIDS 系统较难做到事先中止或防范入侵。而且主机日志中的数据往往是应用活动数据的摘要,缺乏中间数据,影响 HIDS 的检测精度。

由于 NIDS 和 HIDS 系统存在不同的缺陷,仅仅依靠工作在文件和系统命令级的底层操作系统和网络入侵检测系统无法保证检测的效率和精度,因而近年来,通过利用应用系统的独特语义特征检测应用入侵以弥补网络和主机入侵检测系统不足的研究得到重视。本文总结了应用入侵检测技术的研究与进展,指出了应用入侵检测技术研究的难点和需要解决的关键问题,分析和展望了对今后的发展趋势。

2 应用入侵检测系统

由于主机和网络入侵检测系统基于信息系统底层的文件和系统命令级,往往无法检测应用级的入侵或一些权限滥用,如某公司会计利用自己的合法权限擅自将自己的工资提高若干倍,这种内部权限滥用只能通过应用系统本身的语义和结

*)航空科学基金项目(02F52033)、江苏省高技术项目(BG2004-005)。林冬梅 副教授,当前主要研究方向:信息安全、数据库安全;钟 勇 博士,副教授,主要研究方向:信息安全、数据库安全、网络安全;秦小麟 教授,博士生导师,主要研究方向:安全数据库、时空数据库、信息安全等。

构来加以检测，无法从底层的操作系统或网络来检测。又如whisker扫描器能够通过SSL加密传输来扫描Web服务器的CGI脚本的弱点，使得网络级检测无法发现异常。现今的应用系统，如DBMS、Web服务器、邮件服务器等都具有自己的数据结构、应用语义和独特的使用方式，利用应用系统自身特征进行异常检测，能够弥补网络和主机入侵检测系统的不足。

应用入侵检测系统(Application Intrusion Detection System，AppIDS)是近年来由HIDS和NIDS发展而来的。所谓应用入侵检测就是使用应用语义来检测更细微的异常行为，特别是检测内部用户稍微偏离正常行为的滥用[9]。由于AppIDS在应用系统的语义环境中检测异常行为，其检测对象往往是已绕过主机和网络入侵检测系统获得合法身份的内部入侵者或滥用者，而对于其它外部入侵者，则必须依靠主机或网络入侵检测系统。AppIDS应与HIDS和NIDS相互配合，形成完整的入侵检测体系。如AppIDS检测到用户的异常活动，但需要依靠HIDS识别用户的身份，依靠NIDS确定用户的位置和主机地址；相反，当HIDS检测到应用系统突然创建大批量的文件，但无法确定该现象是否合法，需要AppIDS对该现象进行识别。因而，AppIDS是HIDS和NIDS系统的重要补充而不是替代。

3 应用入侵检测技术现状

现有的应用入侵检测技术主要集中在数据库管理系统(DBMS)、Web服务器、邮件服务器以及其它一些应用程序。

3.1 基于数据库管理系统的应用入侵检测

作为信息系统重要数据的存储中心，数据库往往成为最吸引攻击者的目标，因而对数据库入侵检测的研究开展较早。从早期的对数据库推理、存储篡改的检测到当今基于数据库事务、查询模式以及应用语义等检测方法的研究。

数据推理[10]指用户在不存取某些数据的情况下也能推断出这些数据，如在多级安全数据库中用户利用低密级数据或外部知识推理出某些高密级数据，使用推理进攻的往往是具有某些合法权限的内部滥用者。

对数据库的存储篡改(storage jamming)[11]是一种恶意修改数据库中的存储数据以降低数据质量的行为，其目的是以错误或低质量数据误导和妨碍对手的行为。存储篡改是一种内部滥用行为。

数据库具有自己独特的事务处理机制和事务查询语言，对用户使用事务语句的模式进行检测是数据库入侵检测的一项重要内容，如指印(fingerprints)技术等[12]。用户在进行查询的时候往往具有固定或类似的模式，通过对用户的查询频繁项(frequent itemsets)[13]以及应用语义[9]是现今数据库入侵检测研究中的主要方法。

对数据库入侵检测是应用入侵检测中开展较早、较为重要和相对成熟的领域，我们在以前的研究中[14]对数据库入侵检测的研究现状做过全面分析，在此不再重复。

3.2 基于Web服务器和Web程序的应用入侵检测

由于互联网的发展，Web服务器和基于Web的应用程序正在成为新的入侵目标。如利用Web服务器的CGI(Common Gateway Interface)或ASP(Active Server Pages)程序漏洞的入侵、利用Web程序漏洞的蠕虫等。

针对这些入侵，Christopher等[15]提出通过对Web服务器的HTTP请求(request)，特别是GET请求的属性变量作为异常检测的基础，如通过对请求属性长度或字符的分布、属性结构的正则表达式、属性在查询中的出现机率或次序等建立异常模型，将不符合这些模型的请求视为异常。该方法仍旧是基于传统的概率模型，误检率较大且实时检测中对HTTP请求的执行速度也有一定的影响。

Almgren等[16]提出一种分析Web服务器CLF日志的工具，该工具在Web日志条目查找入侵痕迹。但由于该工具不能直接与Web服务器交流，因而只能事后分析而无法中止实时入侵。Emerald架构[6]通过在Apache Web服务器中集成数据收集模块，该模块能够直接抽取Apache的内部数据并提交到入侵检测部件。该方法能够中止实时入侵并能分析更多的数据，但由于该架构直接与Web服务器集成，易影响Web服务器的活动和效率，且安装困难。

Ryutov等[17]提出在Web服务器中集成访问控制和入侵检测，从而使访问政策能够观察和报告异常活动并能在受到入侵时修改系统保护策略。Ryutov提出了在Apache服务器中使用通用授权和访问控制API(GAA-API)，该API能够同时实施基于规则的访问控制和入侵检测。但该方法只适用于已知入侵模型的误用(misuse)检测。

目前对Web服务器的入侵检测系统的研究尚处于起步阶段，相关的技术也处于研究阶段，成熟的商业产品尚未出现。

3.3 基于电子邮件服务器的应用入侵检测

电子邮件服务器的应用检测主要集中在对垃圾邮件(spam)的检测上。该领域是应用入侵检测技术中最为成熟的领域，存在众多的研究技术和应用产品。

基于内容的垃圾邮件检测方法是当今垃圾邮件检测的主流技术，主要包括以下一些方法：

(1) 基于规则的人工方法。如采用黑名单-白名单或手工制订规则的方法。黑名单给出发送垃圾邮件的邮件地址、IP范围或域名等，属于黑名单的邮件就被判定为垃圾邮件。而白名单是合法邮件地址，属于白名单的邮件则被判定为合法邮件。或通过手工建立规则的方法来建立垃圾邮件判定规则。现有的商业产品如摩根实时反垃圾邮件系统等多采用该方式。由于人工建立规则有主观性，该方法易造成大量合法邮件的误判或垃圾邮件的漏判；

(2) 基于规则的机器学习方法。通过机器学习的方法来得到垃圾邮件判定规则，如采用Ripper、决策树、Boosting、粗糙集等机器学习方法。该方法的优点是可以生成易理解的规则，缺点是在规律性不明显的应用领域效果较差[18]；

(3) 基于统计的机器学习方法。如kNN、支持向量机(SVM)、Rocchio、Winnow、贝叶斯、聚类分析等方法。此方法目前尚处于研究阶段，实际应用时仍然有许多工作要做。

文[18]对基于内容的各类垃圾邮件检测方法进行了对比。由于垃圾邮件检测与一般入侵检测不同，如用户宁愿接收更多的垃圾邮件而不能接受合法邮件的错判，也即对正确率的高度要求以及垃圾邮件检测对实时性要求的特点，使现有的基于内容的过滤技术难以满足需求。

针对邮件病毒的问题，Salvatore等[19]提出基于用户行为(Behavior-based)的异常邮件检测方法，该方法从用户的邮件历史记录中分析用户的行为模式，如通讯群体模式(cliques)、信件接受模式等，违反这些模式的信件往往意味着是感染病毒的邮件。该方法能检测到新病毒，但也存在正确率难以满足需求、实时性能较低且需要读取用户的邮件日志等缺陷。

3.4 基于应用调用模式的应用入侵检测

基于主机的入侵检测方法常通过应用程序对操作系统级的系统资源使用模式或系统调用来确定异常行为，然后对应用的内部入侵者来说，他们的行为可能并不偏离该应用对系统资源的使用模式。检测这些内部入侵者需要从应用本身的调用模式入手。

在主机入侵检测中，Forrest 等[20]提出的基于特权进程的系统调用序列来检测入侵的方法是较为著名的算法。在该算法基础上，Jones 等[21]提出可以基于应用程序的语言库调用序列作为应用入侵检测的基础。语言库调用序列由于能够利用应用程序的特定语义，能更面向应用，因而具有更好的性能和检测率。Jones 的实验证明了这一点。

在 Forrest 算法的基础上，Stillerman 等[22]提出在基于 CORBA 的应用中利用客体对象向服务对象发出的请求信息(request messages)序列建立正常模式的方法来检测入侵，他们的方法可以应用在所有基于 CORBA 的应用中。

3.5 基于应用语义的应用入侵检测

应用系统或应用程序往往具有一定的应用语义，而这些应用往往是入侵者所难以掌握的。应用语义的独特性和精确性可以有效地提高应用入侵检测的准确性和粒度。

Rosset 等[23]在研究通讯信息系统的欺诈中利用用户通话记录的语义来检测电话盗用和诈骗等活动，典型的例子如同一个人不可能在某一时间段内同时在相隔较远的两个地方有通话记录等。Stolfo 等[3]对在金融信息系统中利用金融事务的语义来检测信用卡诈骗(Credit Card Fraud)等方法也有较深入的研究。

Sielken[9]提出在入侵检测中使用应用语义并列举了基于应用语义限制和统计的例子。应用语义限制如医生只能查看他所治疗病人的病历、医生开出的处方只能是他专业范围内的处方等，应用语义限制构成基于规则的入侵检测系统；应用语义统计如病人服某种药的次数和剂量应与其它相同的处方之间有一定的相似处、病人购药的订单应大多数发生在白天上班时间等，应用语义统计构成基于统计的入侵检测系统。

由于应用语义的特殊性和难以通过机器学习等自动方法产生，基于应用语义的方法常作为其它应用入侵检测方法的辅助方法。

3.6 基于分布式系统的应用入侵检测

在分布式系统特别是互联网中，由于底层网络和操作系统的不可信，黑客常能利用底层系统的缺陷绕过底层安全系统，中止或控制应用进行，因而分布式关键应用需要有自我防护能力。针对分布式系统不可避免存在突破底层防线的可能性，Webber 等[24]认为分布式应用应具有自我防护能力，因而提出防卫使能应用(Defense-Enabled Applications)的概念。防卫使能应用包括两方面的内容：一是应用能尽量延缓入侵者获取特权的行为，如通过多层次特权和安全域的划分，入侵者要获取最高特权，必须逐级获取各层次和各安全域的权限，从而延缓获取最终特权的时间；二是在受到入侵时应用能做出相应的反应和改变，如改变服务端口、协议，使用后备服务器等。

Schantz 等[25]认为由于应用防卫使能的功能也较为复杂，应用的防卫功能应与应用本身的功能分开，否则易产生代码庞大、难以处理的应用程序。因而，他们建议应使用在应用和底层系统之间的中间件层来实现应用的防卫功能。中间件使防卫机制既能与应用集成又不影响应用原有的功能。

分布式系统中的防卫使能应用只是一个应用检测的概念和实施框架，对具体的应用仍须使用上述的应用检测算法或融合底层入侵检测算法。

最后，由于应用系统中往往存在多种应用，入侵者可能在不止一处留下入侵痕迹，如 Web 服务器、数据库中，因而上述的入侵检测方法可以结合使用。如 Shu Wenhui[26]等设计的应用入侵检测系统中提出使用两层的结构，第一层使用不同应用的检测方法检测各个不同的数据源，如 Web 服务器、Web 数据库，发生异常则产生预警。第二层再将预警合并成报警链，通过检测报警链中是否存在预定义的进攻范式来确定入侵。Stolfo[3]等也提出使用元学习(meta-learning)的方式来集成多种不同的应用检测方法。

4 应用入侵检测系统的实现方法

4.1 实现方式

如图 1[27]所示，按照应用入侵检测系统与应用程序的交流模式，应用检测模块的实现方法有如下 4 种[27]：

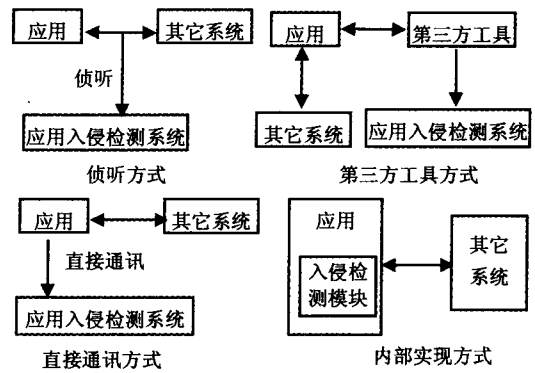

图 1 应用入侵检测系统实现方式

(1)侦听方式。应用检测系统通过侦听应用与其它系统的交流进行入侵检测。侦听方式由于部署容易，且不需要修改和重配置被检测系统，是网络操作系统常用的实现方式。该方式存在的问题是：侦听活动可能影响应用与其它系统传输信息的性能；传输数据必须复制到检测系统中可能影响系统的整体资源利用；检测系统必须有较高的速度来截获应用的通讯；检测系统必须了解应用的传输语义，导致系统更为复杂；应用如果对传输到其它系统的信息加密，将影响检测系统的运行。

(2)第三方工具。应用检测系统利用第三方工具(如日志记录或分析工具)与应用进行间接交流。该方式可以降低对检测系统速度的限制，检测系统可以直接利用第三方工具中的数据而不需要将数据复制到系统中，提高了资源的利用效率。但该方法使检测系统依赖于第三方工具，限制了检测系统的使用范围，并使检测系统性能受到第三方工具的影响。

(3)直接通讯。应用检测系统直接与应用通讯，使得检测系统能够直接进行特征选择(Feature Selection)，减少了应用与检测系统之间的通讯量。如果对应用和检测系统端到端加密，使整个系统更安全，而且检测系统也能直接根据应用程序的结构进行优化(如对结果的预计算或缓存)来提高性能。但直接通讯方式也存在无法获取应用程序部分内部数据以及性能等问题。

(4)内部实现。应用检测系统直接作为应用程序的一部分。与上述外部实现的方式相比，内部实现有较多的优势。由于应用程序本身较为复杂，外部难以获得其内部的大量中间数据。而在应用内部实现则可以获取这些数据，使检测系

统有更高的精确度。而且，应用检测系统与被检测应用集成，能直接分析和选择数据，负载较少，具有更好的性能。另外，由于内部实现方式检测系统与应用程序集成在一起，也易影响被检测系统，更新、修改和升级更困难。表1是Diego对两种方式的优势和问题的部分对比[28]。

表1 内部实现与外部实现比较

实现方式	优势	问题
外部实现方式	易修改，检测系统易在系统中增加或去除。 可使用面向任务的任意语言编写。	检测系统获取数据存在延迟。 数据在检测前易被入侵者修改。 性能影响较大。 对检测数据的获取受到限制。
内部实现方式	获取数据延迟少。 入侵者很难修改被检测数据。 检测系统不容易受到破坏或篡改。 与被检测系统集成，能直接分析和选择数据，负载较少。 能只在需要时运行，可减少CPU占用时间。 数据存取不受限制。	安装需要了解部分被检测系统原代码。 需要对被检测系统进行修改，安装困难。 安装不正确易影响被检测系统 更新、修改和升级更困难。 系统可移植性差

4.2 通讯方式

AppIDS是HIDS和NIDS系统的重要补充而不是替代。由于信息系统活动的多层次性，单一性的检测往往难以达到良好的效果。AppIDS必须能与底层操作系统和网络检测系统集成或数据交换。与底层操作系统和网络入侵检测系统的通讯是一个完善的AppIDS的重要组成部分。

使用审计日志进行信息交流是最简单的通讯方式，但由于存在延迟，难以互操作以及使日志结构复杂化、安全性和通用性等因素，通过审计日志交流的方式只能应用在简单的特定场合。通用入侵检测框架[29](Common Intrusion Detection Framework，CIDF)是一套在入侵检测构件之间进行互操作和信息共享的通用语言、协议和API，该框架由于通讯信息类型的限制和缺乏与TCP/IP的集成协议，不符合近年来标准制定过程中XML(eXtensible Markup Language)语言成为基本信息交换格式的发展趋势。互联网工程任务组(Internet Engineering Task Force，IETF)目前正在制定的相关标准入侵检测信息交换格式(Intrusion Detection Message Exchange Format，IDMEF)和事件对象描述交换格式(Incident Object Description and Exchange Format，IODEF)等相关的系列应用安全标准[30]，将成为入侵检测系统交换的标准格式。未来AppIDS与其它入侵检测系统的通讯应符合这一标准。

结束语和展望 由于NIDS和HIDS系统存在缺陷和不足，使AppIDS的研究受到重视。AppIDS能够利用应用系统自身的数据结构、应用语义和独特的使用方式等特征进行入侵检测，能够提高入侵检测的精度和效率，是入侵检测研究未来发展的重要方向。但AppIDS的研究也面临着众多的问题需要解决。AppIDS需要针对特定应用系统进行设计，而应用程序的多样性使开发通用AppIDS困难。AppIDS与应用系统的紧耦合也对应用系统本身的性能和稳定性造成影响。而且，由于应用系统的易变性，使AppIDS的更新、升级和修改也较传统的入侵检测系统困难。目前，AppIDS的研究和开发仍处于起步阶段，除邮件过滤产品和部分未成熟的产品外，市场上也缺乏完善和实用的商业产品。因此，对AppIDS及其算法的研究不仅在学术上、理论上有很高的研究价值，也有广泛的应用和实践价值。

参考文献

1 Ammann P, Jajodia S, McCollum C D, et al. Surviving information warfare attacks on databases. In: Proc. of the IEEE Symposium on Security and Privacy, Oakland, CA, 1997. 164～174

2 Lippmann R, Fried D, Graf I, et al. Evaluating intrusion detection systems: The 1998 darpa off-line intrusion detection evaluation. In: Proc. of 2000 DARPA Information Survivability Conference and Exposition, Los Alamitos, CA, 2000. 12～26

3 Stolfo S, Fan D, Lee W. Credit card fraud detection using meta-learning: Issues and initial results. In: Proc. of AAAI Workshop on AI Approaches to Fraud Detection and Risk Management, California, USA, 1997. 83～90

4 Axelsson S. Intrusion Detection Systems: A Survey and Taxonomy: [Technical Report]. 99-15. Chalmers University of Technology, Dept of Computer Engineering, Göteborg, Sweden, 2000

5 Allen J, Christie A, Fithen W, et al. State of the Practice of Intrusion Detection Technologies: [Technical Report]. CMU/SEI-99-TR-028, ESC-99-028. Carnegie Mellon University, Software Engineering Institute, January 2000

6 Almgren M, Lindqvist U. Application- Integrated Data Collection for Security Monitoring. In: Proceedings of the Fourth International Symposium on the Recent Advances in Intrusion Detection (RAID'2001), Davis, USA, 2001. 22～36

7 Smaha S E. Haystack: an intrusion detection system. In: Proc. of the 4th Aerospace Computer Security Applications Conference, Washington, USA, 1988. 37～44

8 Denning D E. An intrusion-detection model. IEEE Transaction on Software Engineering, 1987, SE-13: 222～232

9 Sielken R S. Application Intrusion Detection: [Technical Report]. CS-99-17. University of Virginia Computer Science Department , June 1999

10 Yip R, Levitt K. Data Level Inference Detection in Database Systems. In: Proceedings of the 11th IEEE Computer Security Foundations Workshop, Rockport, Massachusetts, 1998. 179～189

11 McDermott J, Goldschlag D. Towards a model of storage jamming. In: Proceedings of the IEEE Computer Security Foundations Workshop, Kenmare, Ire-land, June 1996. 176～185

12 Low W L, Lee S Y, Teoh P. DIDAFIT: Detecting Intrusions in Databases Through Fingerprinting Transactions. In: Proceedings of the 4th International Conference on Enterprise Information Systems (ICEIS), 2002. 264～269

13 Chung C Yip, Gertz M, Levitt K. DEMIDS: A Misuse Detection System for Database Systems. In: The Third Annual IFIP TC-11 WG 11.5 Working Conference on Integrity and Internal Control in Information Systems, 1999

14 钟勇，秦小麟. 数据库入侵检测研究综述. 计算机科学，2004，31(10)：15～18

15 Kruegel C, Vigna G. Anomaly Detection of Web-based Attacks, In: Proc. of the 10th ACM Conference on Computer and Communications Security, Washington D C, USA, 2003. 251～261

16 Almgren M, Debar H, Dacier M. Lightweight Tool for Detecting Web Server Attacks. In: Proceedings of the Symposium on Network and Distributed Systems Security (NDSS '00), San Diego, CA, 2000. 157～170

17 Ryutov T, Neuman B C, Kim D, et al. Integrated Access Control and Intrusion Detection for Web Servers. IEEE Trans Parallel Distrib Syst, 2003, 14(9): 841～850

18 王斌，潘文锋. 基于内容的垃圾邮件过滤技术综述. 中文信息学报，2005，19(5)：1～10

19 Salvatore J S, Hershkop S, Wang K, et al. A Behavior-based Approach to Securing Email Systems. In: Proc. of Mathematical Methods, Models and Architectures for Computer Networks Security, St Petersburg, Russia, 2003. 57～81

20 Hofmeyr S A, Somayaji A, Forrest S. Intrusion Detection Using Sequences of System Calls. Journal of Computer Security, 1998, 6: 151～180

21 Jones Y J, Li Y. Application Intrusion Detection Using Language Library Calls. In: 17th Annual Computer Security Applications Conference (ACSAC'01), New Orleans, Lousiana, 2001. 442～449

(下转第41页)

$x_{ofdf-dfe}$(交换次序后的解)

步骤 2.(交换 H 的列矢量) FOR $i=1:n_T$ $h'_{n_T-i+1}=h_{\pi(i)}$,这里 $H'=[h'_1,h'_2,\cdots,h'_{n_T}]$为调整顺序后的新矩阵。

步骤 3.(QR 分解)$H'=[Q_1Q_2]\begin{bmatrix}R\\0\end{bmatrix}$

步骤 4.(得到 OZF-DFE 解)$y'=Q_1^Ty$。IF 不是一次突发传输内的第一次传输 THEN 利用(14)式递归地得到 $x_{ofdf-dfe}$。

步骤 5.(得到初始半径)$r^2=\|y'-Rx_{ofdf-dfe}\|^2$

步骤 6.(执行 SD 主体)执行算法 1 的步骤 2 到步骤 7,得到解 $x'=(x'_1,\cdots,x'_{n_T})$。

步骤 7.(得到最终解)FOR $i=1:n_T$,$x_{\pi(i)}=x'_i$

需要注意的是,算法 3 在一次突发传输中,步骤 1 到步骤 3 只需要运行一次。

2.2 仿真结果

仿真中,考虑一个 16-QAM 星座的 4×4MIMO 系统,突发长度 L 为 100 个符号周期。信道矩阵元素 $h_{ij}:CN(0,1)$,这里 $CN(0,1)$表示具有单位方差的零均值复高斯分布。噪声的方差为 $\sigma^2=\frac{n_T\bar{E}_s}{2\log_2 Q}10^{-SNR/10}$,这里 $\bar{E}_s$ 为星座点的平均能量,Q 为星座大小。SNR 定义为接收天线上每比特能量同噪声功率谱密度的比值。在我们的仿真环境设置下,$\bar{E}_s=10$,$Q=16$。对应的实模型是一个具有 8 个未知数、8 个方程的方程组。算法的复杂度用算法消耗的 CPU 周期数来衡量。整数加减运算、浮点加减运算、浮点乘法、浮点除法以及求根运算需要的 CPU 周期数分别如文[10]设置为 1、4、7、13 和 20。由于在一次突发的传输中,一些预处理只需运行一次,如 QR 分解、对 H 进行排序等,随着 $L\to\infty$,它们的计算量是可以忽略的,所以这些操作所消耗的 CPU 周期在仿真中没有记入总的周期。

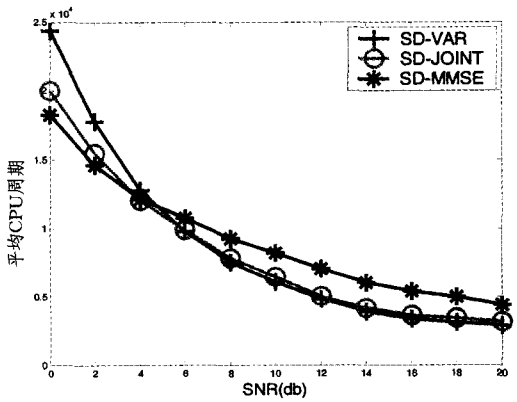

图 1 不同优化策略复杂度比较

我们比较了三种优化策略对 SD 的影响,分别是:初始半径基于噪声方差的 SD(SD_VAR)、初始半径基于 MMSE 解的 SD(SD_MMSE)以及基于本文联合优化策略的 SD(SD_JOINT)。这三种 SD 都使用了在第 2 节提到的对 SD 本身的改进策略。

从图 1 可以看到在低 SNR 范围,基于 MMSE 的优化策略的 SD 具有最低的复杂度,联合优化策略的复杂度相对它稍微偏高。这是由于 ZF 类的检测器使噪声增强造成的。在高 SNR 范围,基于噪声方差的优化策略和联合优化策略具有更低的复杂度。其中联合优化策略和基于噪声方差的优化策略几乎有相同的复杂度。从整个宽的 SNR 范围看,联合优化策略是最优的。如果工作在高 SNR 环境下,由于基于噪声方差的优化策略的 SD 逻辑结构更简单,所以也是一种非常好的优化策略。

结束语 选择合适的预处理方法能够大大地降低 SD 算法的复杂度。本文提出了一种联合优化的预处理策略,通过一个嵌入在 SD 中的次优检测器 OZF-DFE 得到优化的检测顺序和优化的初始搜索半径。仿真表明,在整个 SNR 较宽范围内,这种联合优化策略能够达到最优的复杂度。当系统工作在高 SNR 范围内时,这种优化策略同基于噪声方差的优化策略具有几乎相同的复杂度。

参 考 文 献

1 Foschini G J, Gans M J. On Limits of Wireless Communications in a Fading Environment when Using Multiple Antennas. Wireless Personal Communications, March 1998,6:311～335
2 Wolniansky P W, Foschini G J, Golden G D, Valenzuela R A. V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel. presented at IEEE ISSSE-98, Pisa, Italy, September 1998
3 Fincke U, Phost M. Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. Mathematics of Computation, April 1985,44:463～471
4 Hassibi B, Vikalo H. On the Sphere-Decoding Algorithm I. Expected Complexity. IEEE Transactions on signal processing, 2005,53:2806～1818
5 Qianlei L, Luxi Y. A novel method for initial radius selection of sphere decoding. presented at VTC2004-Fall, 2004
6 Damen M O, Gamal H E, Caire G. On maximum-likelihood detection and search for the cloest lattice point. IEEE Trans. Infom. Theory, 2003,49:2389～2402
7 Viterbo E, Boutros J. A universal lattice code decoder for fading channels. IEEE Trans. Inform. Theory, July. 1999,45:1639～1642
8 Agrell E, Eriksson T, Vardy A, Zeger K. Closest point search in lattices. IEEE Trans. Inform. Theory, Aug. 2002,48: 2201～2214
9 Yongtao W, Roy K. Reduced-complexity sphere decoding via detection ordering for linear multi-input multi-output channels. presented at SIPS 2004, 2004
10 Yang Z, Liu C, He J. A new approach for fast generalized sphere decoding in MIMO systems. IEEE Sig. Proc. Letters, 2005,12: 41～44

(上接第 13 页)

22 Stillerman M, Marchau C, Stillman M. Intrusion detection for distributed applications. Communication of ACM, 1999, 42(7): 62～69
23 Rosset S, Murad U, Neumann E, et al. Discovery of Fraud Rules for Telecommunications - Challenges and Solutions. In: Proc. of Knowledge Discovery and Data Mining (KDD), San Diego, CA, USA, 1999. 409～413
24 Webber F, Partha P P, Richard E, et al. Defense-Enabled Applications. In: Proc. of DARPA Information Survivability Conf. (DISCEX II), Anaheim, CA, 2001. 119～125
25 Schantz R E, Webber F, Partha P P, et al. Protecting Applications Against Malice Using Adaptive Middleware. In: Proc. of International Workshop on Certification and Security in E-Services, Montreal, Quebec, Canada, 2002. 73～108
26 Shu Wenhui, Daniel T. A Novel Intrusion Detection System Model for Securing Web-based Database Systems. In: Proc. of 25th Annual International Computer Software and Applications Conference (COMPSAC01), Chicago, Illinois, 2001. 249～254
27 钟勇.安全数据库异常检测和若干关键技术研究:[博士论文].南京航空航天大学,2006.6
28 Diego Z. Using Internal Sensors for Computer Intrusion Detection:[PhD Paper]. Purdue University, 2001
29 Kahn C, et al. A common intrusion detection framework. Draft submission to a nice publication, 1998
30 Organization for the Advancement of Structured Standards: [Technology Reports]. Application Security Standards. http://xml.coverpages.org/appSecurity.html