一种分布式入侵检测系统构架

A Distibuted Intrusion Detection Frame

韩 宏 卢显良

(电子科技大学计算机科学与工程学院 成都610054)

Abstract As the process of proliferation of the Internet progresses and network security, the intrusion detection becomes a technical attraction. Now, the research of intrusion detection technology is an emerging and thriving academic field. Because of security and the confidential, papers on the topic are rare. Generally, those papers only involve with conceptions and common knowledge. The papers on distributed intrusion detection frame, intrusion detection immune system and resisting subversion are more rare. The paper is on these problems and solves some difficulties.

Keywords Distributed, Intrusion detection, Resisting subversion frame, Intrusion immune system, System interface

1 入侵检测和分布式入侵检测现状及问题

计算机安全技术可划分为主动防御和被动防御两种。后者典型的有诸如防火墙、标识与认证、访问控制等手段。主动防御代表的是入侵检测,它主动实时跟踪系统入侵和误用,及时作出响应,是传统安全手段的很好补充和完善。入侵检测正成为热点,美国政府支持了诸如 EMERALD、NIDES 和 DIDS 等项目。分布式入侵检测因具有较强的容错和抗毁能力、分布式攻击检测能力和开放性架构等优势成为未来研究方向[1,2]。分布式入侵检测要解决以下问题[2]:容错和抗毁、对具体系统能方便裁剪以及系统随时间演化不断优化。

在分布式架构中,当上级节点发生故障时,分级结构容易产生单点失效;纯分布式具有很好的容错性,但因开销和复杂度等因素没有一个实用化系统采用,分级构架系统有 Centrax、NIDES、AAFID、NetStat 等[1]。对于两级结构的 Centrax 和 NIDES,当唯一中心处理单元出故障时,系统将会失效;NetStat 采用底层自治代理提高了生存性,但存在顶层分析器单点失效。AAFID 的三级结构也存在顶层单点失效[3]。关于纯分布式构架,仅见于探索性研究[2]。现有系统均不具备自恢复能力,本文提出了一种具有较强抗毁性的结构。

开放性和伸缩性是一个研究方向[1,7]。入侵检测系统应具备集成各种分析工具的能力,以增进系统性能[4]。AAFID 基于消息的方案由于限制只能用一种语言(Perl)编写,Bro 实现必须用 C++ 扩展[3]。随着网络规模不同,方便地裁剪系统非常必要,本文就此提出了一种方案。

攻击技术变化迅速,系统需要更新检测知识。现有方法都是通过人工添加,速度和质量不理想。在分布式计算环境中,自动发现和共享新知识能极大提高整个系统防御性能[6]。提供合理简洁的知识发现和共享框架使系统具自进化能力是入侵检测研究重点,本文提出了入侵免疫的新概念。

2 系统核心框架

本框架由以下三部分构成:系统检测框架、入侵免疫框架、抗毁性结构框架。

2.1 系统检测框架

本子框架未采用常见固定两级或三级方式,而是提供基本单位由用户任意建立两级或多级的自由方式。检测框架基本单位是检测器和代理探测器。

●代理探测器:是独立运行的自治体,它为系统最底层,功能是收集原始数据作出一定分析和反应并将数据抽象上传父检测器。代理探测器可加快反应速度,减少网络流量,消除中心节点计算瓶颈,同时增加系统抗毁能力。

●检测器:处理来自子代理探测器或子检测器的信息并作出综合分析响应。检测器负责配置和控制所辖探测器与检测器,并执行所辖失效节点的恢复。

两者关系如图1所示。检测器可以是其它检测器的代理,这样能任意组合成多级分布式系统,具有良好伸

韩 宏 博士研究生,研究方向:计算机网络安全、分布对象技术、软件工程。卢显良 教授、博士生导师,研究方向:计算机网络,操作系统。

缩性。

系统将控制和数据传递机制作为共同框架，只要遵循协议，第三方检测软件可无缝集成进系统。框架采用了两种通讯机制：基于消息和基于中间件。前者是通过基本消息格式，以消息传递粘合各模块；后者是定义一套系统 API，通过中间件完成模块间通讯。框架用 Java 实现，JINI 为中间件。非 JINI 实现，可用 JINI 包装(wrap)加入系统。以下简述基于消息的机制。

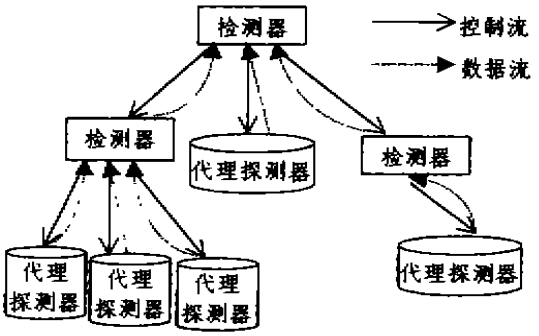

图1 检测子系统构架

消息分为命令和数据。命令分公共和私有命令，公共命令属于框架必须支持的最小命令子集，私有命令是各代理探测器自定义语义。图2是消息格式。

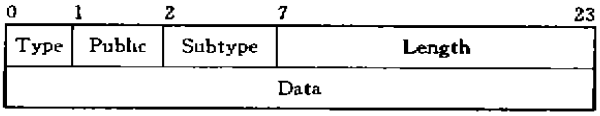

图2 消息帧格式

●Type：数据或命令帧。

●Public：公共或私有命令。当公共时，Subtype 有效。否则私有命令将封装在 Data 中。

●Subtype：公共命令集中子命令。如 Run、Stop 等。

●Length：数据段长度，单位 Byte。

●Data：数据段。

命令消息集合中，现在定义的 Subtype 命令有：

1)Run：启动节点。在 Data 段填入被启动节点的名字和参数。

2)Stop：停止节点。

3)Subscribe：父节点向子节点订购消息，当有满足的消息时，子节点将传回消息。采用"推"的方式可减少通讯量。

4)SetParameters：设置节点参数，包括可能的检测规则等。

5)GetParameters：获取节点参数。

6)Heartbeat：子节点定期播送表示正常的信号。

对私有命令，父节点不需了解其语义。它被封装在消息数据段发送给子节点。这样系统各部分在遵循公共协定完成合作下，可以有更丰富的功能。

数据型消息是子节点"推"给父节点所订购的消息。它包括以下字段：

源地址、目的地址、协议、应用程序、时间、数据选项、数据长度和数据。

以上框架以自由分级结构避免了固定分级方式的问题，使系统具有良好伸缩性。而节点自治性保证了系统的实时性。同时通过基于消息和基于中间件的机制，系统能集成各种检测软件，使其具有良好开放性。

2.2 入侵免疫框架

当今计算机免疫研究成为焦点，美国 DARPA 成立了专项研究；IBM 也成功地将生物免疫机制引入病毒防治，针对入侵手段层出不穷，检测知识严重滞后[3]，我们提出入侵免疫概念：检测知识相当于抗体，免疫框架自动对新异常行为生成新检测知识并自动在系统范围内共享，共享相当于免疫系统将抗体送到被侵害和潜在被侵害机体。入侵免疫框架如图3所示。

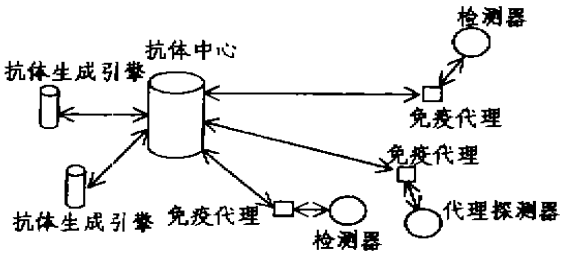

图3 免疫框架

框架由抗体中心、抗体生成引擎、免疫代理和检测节点组成。检测节点是检测器或代理检测器。免疫代理收集检测节点生成的事件数据并送至抗体中心，代理节点向抗体中心订购知识并更新节点知识。抗体中心是检测规则知识库，负责将订购的知识回送到免疫代理，同时管理免疫代理采集分析所需数据，把数据提交抗体生成引擎。抗体生成引擎用数据挖掘方法分析事件数据。它自动生成检测规则和知识，将新检测知识送回抗体中心。框架工作过程如下：

1)检测节点通过免疫代理注册到抗体中心，说明所需知识类型。

2)抗体中心定时从免疫代理收集事件数据，并分类提交给不同抗体生成引擎。采集数据时，根据网络流量作出采集量的动态调整，以免造成过大开支。

3)不同的抗体生成引擎负责不同类型检测知识生成。收到事件数据后，它们分析并生成新规则，送回抗

体中心。

4)当新检测知识形成后,抗体中心将它们及时推送给订购知识的免疫代理,由免疫代理将知识加入检测节点。

手动加入的新知识也可被自动推送到检测节点。系统的配置相当简单:抗体中心和抗体生成引擎在专用的主机上运行,要加入入侵免疫系统的检测节点只需在所在主机安装免疫代理。

2.3 抗毁性结构框架

抗毁性中失效模块的自动重构,国内外尚未见研究报道,但均认同是一个重要课题。限于篇幅,简述结构和协议。

2.3.1 抗毁性结构 关于抗毁性的总体思路是:当系统中某一节点失效,系统自动寻找合适冗余节点启动,并将失效点系统配置转移到新启动节点。此恢复体系中不存在单点失效,对系统多点攻击,也能快速对系统拓扑重构。系统利用检测体系自然的树型结构,让父节点管理子节点恢复。树的根节点、由 N 个恢复精灵组成的恢复域负责其恢复。系统预先配置检测节点冗余备份。守护进程负责启动新节点。抗毁结构如图4所示。

它由恢复域、恢复精灵、守护进程,检测节点和配置仓库组成。下面就抗毁框架功能分述结构各要件。恢复失效节点分两大方面:冗余节点的启动和失效节点等效运行环境配置。

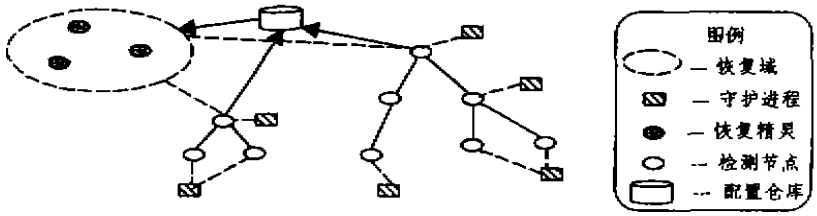

图4 抗毁性结构

冗余节点启动 图4中父检测节点监视子节点的生存。当子节点死亡,父节点查找合适冗余节点并向其所在主机守护进程发命令启动冗余点。根节点的失效恢复统一由恢复精灵组成的恢复域负责。不论树中任意个节点失效,总有其父节点自动恢复;检测树根由不存在单点失效的恢复域处理。执行恢复功能的节点,必须有冗余节点位置分布及类型等信息,我们称其为冗余节点信息图。配置仓库存放最新冗余节点信息图、节点和恢复精灵通过订购在其本地存储。因配置仓库是系统唯一潜在单点失效、所以冗余信息应从本地读取,同时加快处理。信息图获取方式为:检测树根和恢复精灵向配置仓库订购。当有了信息更新,配置仓库把更新推送到订购节点。冗余节点信息图在检测树中由父节点向子节点推送。

节点等效运行环境配置 找到合适冗余节点启动只是恢复过程的第一步。要尽可能还原系统受损前的状态,必须将失效节点运行配置诸如:命令参数、规则集和反应方式等要素,还原到新启动节点。配置仓库存储检测树完整运行环境配置信息。节点配置变动上传父节点,父节点存储所辖节点配置信息,依此层层上传。恢复精灵从配置仓库获得所需信息并本地存储。

系统唯一潜在单点失效——配置仓库失效只影响配置期,管理员可立刻解决。配置完毕仓库将不加入检测工作(恢复决策使用本地信息)。所以它对系统抗毁性、生存性影响很小。

2.3.2 抗毁协议 协议包括:冗余节点变动时算法、检测节点配置变动时算法和检测节点失效时算法。前两个算法是保证系统拓扑配置及冗余信息同步。限于篇幅仅简述检测节点失效时算法。

图5为恢复算法状态转移图。系统平时在正常状态。当受攻击等原因节点失效时、系统转到失效状态。通过恢复精灵或父节点启动冗余节点,系统转入回收状态。启动分两种可能:一、根节点失效时、恢复精灵通过分布式仲裁选出一个精灵负责启动;二、非根节点失效时,由父节点启动冗余点。启动前把失效节点配置传到被启动点所在主机。新启动节点必须回收失效点的子节点,使其成为自身子节点从而恢复系统原树状结构。这时系统在回收状态。当新节点发现欲回收子节点失效,系统转为失效态,新节点将启动冗余点恢复失效子节点。系统可能会启动重复节点,发现后将进入仲裁态消除重复节点。当启动者通过心跳信号中节点 ID 发现有重复节点、或回收子节点时子节点发现多个要求成为其父节点的请求、系统转入仲裁状态。通过仲裁系统转向回收态,当正常回收完毕,系统回归到正常态。失效的检测点将被恢复,系统完成了重构。恢复协议关键是节点回收和重复节点仲裁、它们重构了失效前的树结构。

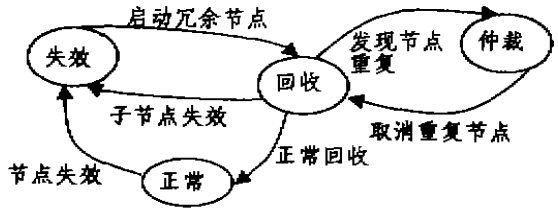

图5 系统恢复的状态转移图

仲裁算法的关键是优先权决定。通过集中分配会造成单点失效，分布式又存在一致性问题。我们采用计算机的地址，比如 IP 地址决定优先权。它具有唯一性，又不存在集中分配和分布式的缺陷。简述如下：恢复精灵监听根节点心跳(精灵和根节点在同一网段，根不知道精灵的位置，它的心跳是广播消息)，当没有心跳后，精灵广播自己 IP 地址表明要启动节点，收到消息的精灵比对自己的地址，如自己地址值大于消息中给出的就回送自己地址。发送者在时限内收到比自己地址值高的回复，就放弃启动企图。当产生重复节点时也是靠网络地址值高低分布式地决策出将关闭的节点。篇幅所限不能详述。

结论 本文研究了分布式入侵检测的问题和现状，提出了解决方案。可任意扩展的分布式结构，弥补了两层和三层模式伸缩性差及存在单点失效问题。基于消息和基于中间件的机制，使系统有良好开放性，方便集成第三方检测工具。文中提出的入侵免疫框架和抗毁框架在这一领域未见报道。入侵免疫框架通过模拟生物免疫机制自动发现共享新检测知识使系统自我进化。抗毁性框架针对安全系统的生存性要求提出，文献中广泛指出这一课题的重要性，但尚无专门研究。本文提出了框架并解决了系统重构中的若干问题。在这一领域有积极意义。

参 考 文 献

1 Allen J, Christie A. Sate of practice of intrusion detection technologies: [Technical report CMU/SEI-99-TR-028]. 1999
2 Janse W, Mell P. Applying Mobile Agents to Intrusion Detection and Response: [NIST Interim Report(IR)-6416 Oct]. 1999
3 Paxson V. Bro: A System for Detecting Network Intruders in Real-Time. Computer Networks, 1999, 31(23-24): 2435～2463
4 Porras P A. Detecting Computer and Network Misuse Through the Production-based Expert System Toolset(P-Best). In: Proc. of 1999 IEEE Symposium on Security and Privacy, May 1999. 9～12
5 Spafford E. An Architecture for Intrusion Detection Using Autonomous Agents: [COAST Technical report]. June 1998
6 Lee W. A Data Mining Framework for Building Intrusion Detection Models. In: Proc. of 1999 IEEE Symposium on Security and Privacy, May, 1999
7 Jagannathan R. Lunt T. System Design Document: Next-Generation Intrusion Dectection Expert System (NIDES). System Design Document: A007, A012, A014 March 9, 1993
8 Ingram D J. Kremer H. Distributed Intrusion Detection for Computer Systems Using Communicating Agents. In: the Proc. of the 2000 Command and Control Research and Technology Symposium, June 2000
9 Helemer G G., Wong J S. K. Intelligent Agents for Intrusion Detection
10 Libicki L. Defending Cyberspace and Other Metaphors

(上接第129页)

$|j_{x1}\rangle$、$|j_{x2}\rangle$，使 $f(j_{x1})=f(j_{x2})$，但这并不排除存在一不为0的 s，使 $f(j_x)=f(i\oplus s)$，当然，$|i\rangle$不在所找的 $n+\delta$ 个不同的 $|j_x\rangle$中，这时 f 应是第二种情况，但我们会把它判断为第一种情况，出现判断失误。值得庆幸的是，这种可能性只有 $1/2^{n+\delta}$。这样小的可能性的错误在一般情况下是可以允许的。和前一个问题一样，当 N 很大时，量子计算的优越性是很显然的。

结束语 本文分三部分讲述了量子计算的计算能力，第一部分简单介绍了量子计算的基本概念及量子计算具有强大计算能力的原因。第二、三部分举例说明了量子计算具有强大计算能力的原因。量子计算还是计算机科学的一个新领域，希望本文的介绍能让读者建立一个量子计算的简单概念。

参 考 文 献

1 苏月琼. 量子计算机前途无量. 计算机世界，2000(17)
2 成就于后摩尔定律时代的量子计算. 计算机世界，2000(50)
3 Preskil J. Quantum Information and Computation. Lecture Notes