基于 Mobile Agent 的入侵检测架构①

An Architecture of Intrusion Detection Based on the Mobile Agent

刘　凯　宋存义　（北京科技大学 环境工程学院 100083）
周贤伟　（北京科技大学 信息工程学院　100083）

摘要：本文针对在物理上与 Internet 网完全隔离的计算机网络应用环境，提出基于 Mobile Agent 的多层次入侵检测架构，利用自组织映射网络方法，在不同层次的 Agent 中建立二维网格的自组织映射网络模型，分别检测目标系统不同层次上的异常情况。

关键词：入侵检测　自组织映射网络　移动代理　网络安全

1　引言

本文所提出的异常检测方法，是以一个特定的应用环境为背景，在该环境下存在如下特点：

（1）攻击者的攻击目标相对单一（主要是机密数据的窃取）；

（2）可能攻击类型和方式大大减少（DoS 类攻击一般不会出现）

（3）所有用户及系统是可管理对象。本文依据应用环境的特点，提出了基于 Mobile Agent[1] 的多层次入侵检测架构，采用了自组织映射（Self – Organizing Maps）[2] 神经网络方法，学习和捕捉用户和系统的正常行为和状态轮廓特征，以此来发现异常的行为和状态。

2　体系结构

本文所针对的检测目标是一个是物理上与外界隔离的 Intranet，在该环境下的入侵者主要是内部人员，其主要的攻击目标是非授权的数据访问。由于网络的带宽高且采用 Switch 技术，使基于网络的入侵检测方式难于实施。同时，由于网络内的所有主机系统在可管理范围，采用基于主机的入侵检测方式较为适合。考虑到每一台用户主机都是入侵的开始位置，对每一台用户主机进行监测很有必要。此外，由于系统主机（应用服务器、数据库服务器）是入侵的目的位置，也需要对这些主机进行监测。

图 1 给出了基于 Mobile Agent 的 IDS 的系统结构图。该 IDS 系统通过 Mobile Agent 对各主机系统进行监测。系统分三个层次：控制和管理中心、分析器和

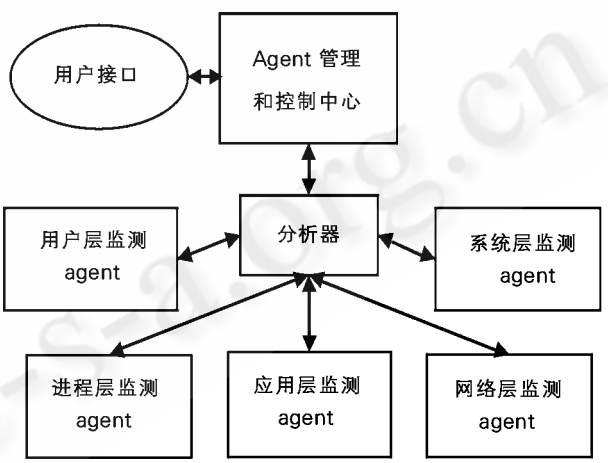

图 1　基于 mobile agent 的 IDS 的结构图

Mobile Agent。控制中心负责协调控制整个 Intranet 网络系统的分析器。分析器可根据主机的业务性质进行配置，比如，对于用户主机，一个分析器可负责某一部门的用户主机的管理工作；对于数据库、应用服务器主机，一个分析器可以负责一台主机的管理工作。分析器负责管理下属的各个 Agent，包括创建、删除、启动、停止 Agent，生成并配置每个 Agent 的检测模型及运行参数，接收并上报 Agent 提交的检测结果。各类不同

① 国家自然科学基金资助项目（66272011）

的 Agent 完成具体的检测任务并负责为分析器提供生成检测模型所必须的数据。

各 Agent 负责不同层次上的数据的收集和检测。对于不同的类型的主机,数据的类型会有所不同,但是基本上可按层次划分如下:

(1) 用户层。包括用户名、用户的登入和登出时间、访问的资源和目录、使用的软件、所用命令等。

(2) 进程层。包括进程的名称、进程的数量和类型、进程之间的关系、进程的运行时间、进程的当前状态、不同进程的时间比例等。

(3) 应用层。针对不同的应用,所监测的数据有所不同。比如对于 Web 应用,我们可以采集 HTTP 访问的特征信息,包括访问时间、源 IP 地址、目的 IP 地址、请求页面、特殊字段等。

(4) 系统层。包括每个用户 CPU 的累计使用时间、虚拟和实际内存使用量、Unix 主机当前可用的磁盘交换空间的大小、可用内存大小、I/O 及磁盘使用情况等。

(5) 网络层,包括当前连接数量和连接状态,用户端和主机端接收和发送数据包的数量、连接的持续时间、连接类型、所用协议和端口等。

目前,采用 Mobile agent 实现入侵检测还是一个较新的方法,但它有良好的发展前景。MA 的一些固有的特点,如自主性、灵活性、平台独立性、协作性等,赋予了入侵检测系统许多新的特性[3]我们提出的入侵检测系统架构,充分利用了 Mobil agent 许多优点同时,也充分考虑到了特殊应用环境下的特点。主要有以下方面的考虑:

① 由于系统面临来自内部用户的攻击,因此可以把监测的目标从系统、应用和数据库服务器主机延伸至用户的主机系统上。

② 由于应用环境具有多个操作系统平台,所以要求 IDS 能跨平台操作。

③ 由于网络拓朴和系统配置会随着应用需求的变化而改变,所以要求 IDS 具有自适应能力。

④ 在占用较少系统和网络资源的同时,能够得到可以接受的虚警率和检测率。

⑤ 通过不断对历史经验的学习,不断改善系统的检测能力。

⑥ 在允许的时间范围内,检测异常事件并及时采取响应措施。

⑦ 系统自身必须简单、安全,尽可能避免新的弱点的出现。

⑧ 避免出现因单点故障而导致系统崩溃的情况出现。

在系统中采用 Mobile Agent 可以大大提高系统的灵活性、可扩充性、自适应性、健壮性等,但是并不能增加异常检测的精度,真正对检测精度起直接作用的是目标系统特征数据的选择和数据分析模型的建立。

3 检测模型

要完成自动化的异常入侵检测工作,首先要自动生成和建立检测模型。根据我们所面对的计算机网络环境,用户主机是我们检测的重点,因为这是入侵者的入口处。其次,我们也要对重要的数据库主机、应用主机保持监测,因为这是入侵者的攻击目标。由于一些主机的操作系统和主要应用有所不同,所采集的数据也会有所不同。除了应用层对不同的应用有较大的依赖外,其他各层的数据通过操作系统的相关日志、系统调用和系统命令可以得到。系统通过在所抽取的各层次上的特征数据上的训练和学习,可以自动产生检测模型。

学习分为二种型式,一种是监督学习(Supervised Learning),另一种是无监督学习(Unsupervised Learning)。本文采用的是 Self – Organizing Maps(SOM)非监督学习算法。因为:

(1) 更有可能检测到新的攻击;

(2) 对输入数据的学习不需要先验的知识;

(3) 能根据环境的变化而自动调整。

SOM 是采用竞争学习的一种神经网络,它能捕捉到包含在输入空间中的重要特征,并通过网络拓扑提供一个结构化的表示。从网络结构上来说,其最大特点是输出神经元被放置在一维、二维或更高维的网格节点上。

本文采用的是最为普遍的二维网格模型,在不同层次上建立相应的 Maps。假定每个 SOM 在输入层接受输入向量 $X=[x_1,x_2,\circ\circ\circ,x_N]^T$。输出层包括网络的 M 个神经元,输入节点通过权值 $wij(i=1,2,\cdots,M;j=1,2,\cdots,N)$ 和神经元相连。每个神经元的权值向量为 $Wi=[w_{i1},w_{i2},\cdots,w_{iN}]^T(i=1,2,\cdots,M)$。利用 SOM

的学习算法,在训练阶段,可以在正常样例中产生聚类(cluster)。该聚类用来决定在使用阶段新样例的正常与异常。本文采用距离函数 $f_d(s,K)$ 来测量新样例 s 接近 cluster K 的程度,并以此来决定新样例是否异常。具体表示如下:

$$f_d(s,\text{Normal}) = \min\{f_d(s,K_i) \mid K_i \in C\}$$

$$x_{\text{abnormal}}(s) = \begin{cases} 1 & \text{if } f_d(\text{Normal}) \geqslant t \\ 0 & \text{otherwise} \end{cases}$$

上式中,C 表示正常子空间的集合。t 用于区分正常类和异常类界线的阈值。式中的 f_d 采用欧氏距离:

$$f_d(s,K) = \| s - w_k \|$$

为了得到 SOM 所要求的输入向量,无论采用何种方式得到的数据,都需要将它们转换为数字表示的某种格式数据。此外,输入参数的变化范围也要进行标准化处理。比如在用户层,对于用户名,可以根据所在的组和授权的级别高低,赋予相应的数据编号。对于所访问的资源和目录,也可以根据所设置的访问权限、安全级别转为相应的数字量。由字符串的数据类型转换为数字量是一个复杂的工作,这一步工作完成的好,才能充分利用 SOM,完成对目标系统的异常检测。

4 系统能力分析

本文只是提出在特殊应用环境下的 IDS 的一个实现架构,系统尚未完全实现。但是我们可以预测,它能检测出许多入侵行为。比如在用户层,可以发现用户的登录失败、越权访问等问题;在进程层,可以检测到不正常的进程内存的分配、优先级和 CPU 的使用等问题;在应用层,可以检测到异常的配置、参数方面的异常;在系统层,可以检测内存、CPU 与外设的异常使用情况;在网络层,可以检测端口的异常使用、数据包数量的异常变化等。

对目标系统资源占用的多少,也是系统能力大小的一个反映。由于 Mobile agent 要在不同的主机间移动,Agent 本身担当数据采集、数据预处理及 SOM 的解算任务,对目标系统的网络和系统资源有一定的占用。要减少资源占用的比例,不影响正常的应用,一方面要使 Agent 保持瘦身,另一方面要降低 Agent 的数据采样和处理频率。本文采用的多层次的数据采集和处理,使每个 Agent 所需处理的数据量大大减少,处理速度也会大大提高。至于采样和处理频率,考虑到在特殊的应用环境中,有较多的防护措施,攻击者要完全达到目的,是要花费一定的时间代价的,所以 Agent 的数据采样和处理间隔时间可以选择的长一些。

由于系统和应用会发生一定的变化,系统需要通过对变化后的样例的学习,产生新的聚类。后一个训练阶段的起始时间,与相邻前一个训练阶段的起始时间的间隔,不能太小,否则会影响目标系统的正常应用。经过对目标系统有关数据的连续三个月的观测,发现数据相当稳定。初步考虑可以让 IDS 系统每隔一个月左右的时间,进行一次新的学习和训练。当然若已知系统和应用发生了变化,可以随即进入新的训练阶段。

检测精度是 IDS 的最重要的技术指标。为了防止系统出现过多的误警,可以将系统主机的检测结果与用户主机的检测结果,进行相互关联和认证。此外,在特定的网络应用环境中,用户、系统和应用相对稳定,攻击者的类型也相对单一,所以用户行为和系统状态的"正常轮廓"能更好地满足检测的要求,预期可以得到更好的检测效果。

5 结束语

本文所提出的基于 Mobile Agent 的入侵检测架构,采用了分层 Agent 的结构,大大减小了采用 SOM 神经网络方法的计算量。但是该架构忽略了各层次间的关联问题。在分析器中建立一个层间关联数据分析模型,为异常检测做进一步的修正和补充,是很有必要的。这是下一步要做的工作。

参考文献

1 Kohonen T. Self – Organizing Maps. Springer, Berlin,1995.

2 张云勇、刘锦德 编著, 移动 Agent 技术, 清华大学出版社,2003.9。

3 GuyHelmer, Johnny S. K. Wong, Vasant Honavar, Les Miller. Lightweight Agents For Intrusion Detection. November. 2000.