MADIDS:一种基于移动代理的新型分布式入侵检测系统*)

罗光春 卢显良 李 炯 张 骏
(电子科技大学信息中心 成都610054)

MADIDS:A Novel Distributed IDS Based on Mobile Agent

LUO Guang-Chun LU Xian-Liang LI Jiong ZHANG Jun
(Information Centre of UEST of China,Chengdu,China 610054)

Abstract When traditional Intrusion Detection System(IDS) is used to detect and analyze the great flow data transfer in high-speed network,it usually causes the computation bottleneck. This paper presents a new Mobile Agent Distributed IDS(MADIDS) system based on the mobile agents. This system is specifically designed to process the great flow data transfer in high-speed network. In MADIDS,the agents that are set at each node process the data transfer by distributed computation architecture. Meanwhile by using the reconfiguration quality of the mobile agents,the load balance of distributed computation can be dynamically implemented to gain the high-performance computing ability. This ability makes the detecting and analyzing of high-speed network possible. MADIDS can effectively solve the detection and analysis performance bottleneck caused by the great flow data transfer in high-speed network. It enhances the performance of IDS in high-speed network. In this paper,we construct the infrastructure and theoretical model of MADIDS,and the deficiencies of MADIDS and future research work are also indicated.

Keywords Mobile Agent,Distributed IDS,Performance

1 引言

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来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,起着主动防御的作用,是网络安全中极其重要的部分。入侵检测系统(Intrusion Detection System,IDS)是识别针对计算机系统和网络系统,检测外界非法入侵者的恶意攻击或试探,内部用户越权使用权限的非法行动的软件系统[4]。

一个典型的入侵检测系统一般包括如下几个部分:事件产生器(Event generators),事件分析器(Event analyzers),响应单元(Response units),事件数据库(Event databases)[5]。几个部件之间通过通用入侵检测对象(Generalized Intrusion Detection Object,Gidos)来交换数据。事件产生器从环境中获取感兴趣的信息,并将其转换成为标准格式;事件分析器分析数据,并产生Gidos;响应单元对Gidos进行处理;事件数据库存储事件和Gidos。

随着计算机界对网络安全问题技术的深入研究,入侵检测技术发展非常迅速。当前,入侵检测系统面临的最主要挑战有两个:一个是虚警率太高;一个是检测速度太慢。随着网络硬件条件的不断发展,检测速度的问题愈发突出。目前大多数入侵检测系统在不牺牲检测质量的前提下尚不能处理百兆网络满负荷时的数据量,而千兆网络还是个不可企及的目标。

为解决检测速度的问题,学术上提出了负载平衡、分布式结构等几种解决方案。采用分布式结构将检测工作进行分布式计算是一个比较可行的方案。

一般认为,分布式入侵检测系统是解决检测速度问题的根本解决途径,因此,分布式入侵检测系统成为网络安全技术研究的热点,而且也研制出一些实验性的分式式入侵检测系统。现有的分布式入侵检测系统的结构是,分布式的采集数据,并将数据交给中央处理台,由其进行集中处理。这种结构有效地解决了高速网络中的数据收集工作。文[3]描述了一种分布式入侵检测系统NetNumen的结构。系统由控制台(Console)和分布式的监控器(monitor)组成。多个monitor分布在网络的各个广播域,通过TCP协议向Console发送告警信息。Console配置monitor的攻击特征表、告警信息过滤表等信息,对各monitor的告警信息进行统一的处理。

现有的分布式入侵检测系统大多存在如下的缺陷:(1)数据传输负荷过大。分布式数据采集器与中央处理台之间的数据交换造成网络带宽的极大浪费,在网络负载较大的时候,特别是大规模、异质网络基础上发起的复杂攻击行为时,数据传输的负荷将使得网络不堪重负[1]。(2)中央处理模块的计算瓶颈。分布式数据采集组件采集的数据均交给中央处理模块计算,处理能力将受制于中央处理计算机的软硬件条件,无法扩展其计算能力。同时,收集和存储入侵信息也将是中央处理模块的极大瓶颈[1]。(3)网络传输的时延问题。由于采集的数据传输到中央控制台普遍具有时延,特别在大规模异质网络中,到达中央控制台的数据反映的是数据被收集时刻的状态,而不能反映网络的当前状态。这些基于过时信息做出的判断的可信度大大降低[4]。并且,使得收集入侵相关信息的工作非常困难。

2 MADIDS的体系结构

为了解决系统通信负载,特别是高速网络中通信负载的问题,我们提出了基于移动agent的分布式入侵检测系统(Active Agent Distributed IDS)。一般来说,agent是指一个能在特定环境下连续、自发地实现功能,并且是与相关代理和进程相联系的软件实体[2]。MADIDS系统包括事件产生agent、事件分析agent、事件跟踪agent、agent服务器几个部分组成,如图1所示。

*)本文受国家九七三支持,项目号973-1-4-2.罗光春 博士研究生.卢显良 教授,博士生导师.李 炯 讲师.张 骏 讲师.

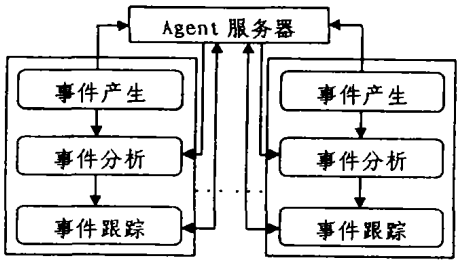


图1 MADIDS 系统的组成

在 MADIDS 系统中，主要包括如下几个部分：

(1)事件产生 agent：事件产生 agent 分布在网络的各个位置，负责收集感兴趣的数据，并且根据 agent 自身的负载和网络的负载情况，将部分数据提交给自身的事件分析 agent 处理，将另外一部分处理不了的数据提交给 agent 服务器，由 agent 服务器分派给其他事件分析 agent 处理；(2)事件分析 agent：事件分析 agent 接受从本地的事件产生 agent 和 agent 服务器的事件分析请求和分析数据，对数据进行分析，并根据自身的负载将处理结果分派给本地的事件跟踪 agent 或者 agent 服务器；(3)事件跟踪 agent：事件跟踪 agent 接受从本地事件分析 agent 和 agent 服务器的事件跟踪请求，对入侵进行跟踪，并在网络负载允许的时候将数据提交给 agent 服务器；(4)agent 服务器：agent 服务器是各个 agent 的中央管理机构。agent 服务器主要完成如下几项工作：a. 接收事件产生 agent 的数据，并将分析任务分派给合适的事件分析 agent；b. 接受事件分析 agent 的数据，并将跟踪任务分派给合适的事件跟踪 agent；c. 监视每个 agent 的负荷，动态地对 agent 的负荷进行平衡；d. 接受事件跟踪 agent 返回的跟踪数据，并将其存储到事件服务器中。

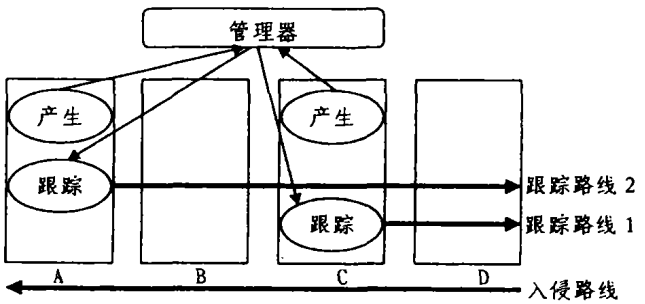


图2 事件跟踪 agent 冲突

与普通分布式 IDS 系统相比，MADIDS 系统具有如下的优点：(1)网络负荷较小。事件产生 agent、事件分析 agent 在得到处理数据的任务之后，优先将数据交给就近的 agent 处理，而不需要交给服务器，再由服务器交给 agent。只有在本地 agent 的负载超过了指定的阈值，处理数据才会传输给 agent 服务器分配。这样就大大减少了大量分析、跟踪数据在网络上传输时对网络造成的负荷。(2)分布式计算。整个入侵检测任务动态地分配到了每个 agent 上，这就避免了在对数据进行处理集中的模式下，中央处理器的计算能力对检测能力的限制。(3)时效性较强。在最优的情况下，agent 的数据处理均在本地处理，能够即时发出跟踪指令；在最差的情况下，agent 的数据处理均需要通过服务器分配计算任务，这种情况的时效性与分布式 IDS 的情况类似。一般情况下，时效性介于最优和最差之间，总体性能得到了提高。

3 agent 协调模型

在 MADIDS 中，agent 服务器将向目标系统分派事件跟踪 agent，并且由于事件跟踪 agent 的移动具有独立性，因此它在目标系统间的移动与其他的事件跟踪 agent 无关。许多事件跟踪 agent 可能跟踪同一个入侵，agent 服务器并不能集中控制各跟踪代理的移动。例如，假如用户 X 远程地按 A-B-C-D 的顺序登录到系统 A、B、C 和 D 中；用户 X 破坏了以上系统，在目标 D 和 B 上分别检测到了 MLSI(Marks Left by Suspected Intruders)；D 和 B 的事件检测 agent 独立地向 agent 服务器发出报告，agent 服务器向 D 和 B 各分派一个事件跟踪 agent；事件跟踪 agent D 代理(Dag)按 D-C-B-A 的顺序来跟踪入侵路由；而事件跟踪 agent B 代理(Bag)则按 B-A 的顺序进行跟踪；因此两个事件跟踪 agent 在 B-A 这一段网络上所做的工作就是重复交叉的。

因此，需要建立一个 agent 的协调模型来协调 agent 之间的工作。所谓协调，就是将不同的活动体组织在一起的构建程序的过程[6]。协调模型需要解决 MADIDS 系统中事件产生 agent 之间、事件分析 agent 之间、事件跟踪 agent 和这些 agent 与 agent 服务器之间的符号信息交换和协议交互的问题。

协商模型可以分为控制驱动和数据驱动两类[7]。控制驱动模型适用于控制流和依赖关系需要规范化的系统，而数据驱动模型适用于自治的、相互不了解的 agent 之间实现互操作的开放应用。MADIDS 系统需要规范化事件分析 agent 之间控制流，应采用控制驱动模型。

下面用 MANIFOLD 语言来描述 agent 服务器与事件分析 agent 之间的协调模型。

```
manifold PrintUnits()import.
manifold ReadFile(port in)import.
manifold Sorter import.
manifold Main()
{
auto process read is ReadFile("unsorted").
auto process sort is Sorter.
auto process print is PrintUnits.
begin:read→sort→print.
}
manifold AtomicSorter(event)import.
manifold Merger()import.
maniford Sorter()
{
  event filled,flushed,finished.
        process atomsort is AtomicSorter(filled).
        stream reconnect KB input→*.
        priority filled<finished.
        begin:
        {
          activate(atomsort),
          input→atomsort,
          guard(input,a_everdisconnected!empry,finished),
          guard(output,a_everdisconnected,flushed),
        }.
        finished:
        {
          ignore filled.
          begin:atomsort→output.
        }.
        filled:
        {
          process merge(a,b|output)is Merger.
          stream KK*→(merge.a,merge.b).
          stream KK merge→output.
          begin:
          {
            activate(merge),
            input→Sorter 0 merge.a,
            atomsort→output
          }
          finished:
        }.
        end:
        {
```

```
        begin:terminated(void).
        flushed:halt.
    }.
}
```

上述程序是agent服务器与事件分析agent协调模型的高度抽象。描述了agent服务器如何智能化地分配事件分析任务，而事件分析agent又是如何智能化地获取事件分析任务的。

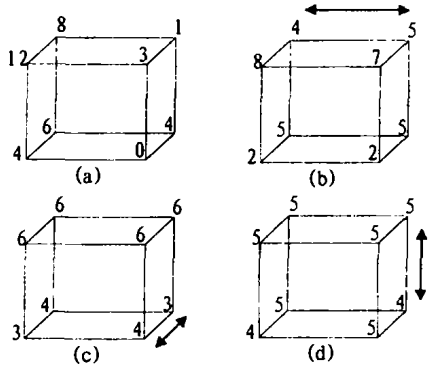
4 动态负载分配

agent服务器能够动态对负载进行分配，这是MADIDS系统最大的特点之一。分配的负载包括事件分析数据和事件跟踪数据，agent服务器在运行时根据各个agent的负载，动态地分配事件分析和事件跟踪的数据，充分利用全系统的所有资源，以提高整个MADIDS系统的性能。

采用维交换DE算法实现agent服务器的负载平衡分配。系统中的每个agent都有地址 $u_n u_{n-1}\cdots u_1$，$u_i \in \{0,1,*\}$，$1 \leqslant i \leqslant n$，$u_i$ 是地址的第 i 维。定义 $u_i \in \{0,1,*\}$，* 表示1或0，这样可以用 $u_n u_{n-1}\cdots u_1$ 既表示agent，也表示agent组。符号 $u^{(d)}$ 表示节点 u 在第 d 维的邻接agent，L(u)表示agent的负载。地址中有偶数个1的agent是偶agent，反之，是奇agent。

agent服务器可以采用基于无错超立方体的agent负载分配算法来分配负载。步骤如下：(1)选择一个维度 d，通过沿 d 维的一个回合的维交换，使得每两个相邻agent的 u 和 $u^{(d)}$ 负载平衡。结果是偶agent(u)的负载为 $\left\lfloor \frac{L(u)+L(u^{(d)})}{2} \right\rfloor$，奇agent($u^{(d)}$)的负载为 $\left\lceil \frac{L(u)+L(u^{(d)})}{2} \right\rceil$。(2)在任何顺序在剩下的 n−1维上重复步骤1。

图3显示了在3维无错立方体上，通过3次维交换，实现负载平衡。维交换DE算法是分布技术中的一个重要理论，其有效性从理论上[9]和应用上[10]都可得到验证。



(a)初始负载分布　(b)第一维交换
(c)第二维交换　(d)第三维交换

图3　三维无错立方体的维交换

以上算法的有效性在文[3,12]中已得到证明。

5 系统可重构性

重构是移动agent的重要性质，本文使用负载平衡的策略来平衡检测计算工作，而这可以用移动代理机制的重构能力来加以实现。本文所指的重构是指：agent在分布式检测系统中的动态重新布置。

重构的过程是：首先当一个正在执行的agent想要将自己外送到远端时，会对agent服务器发出请求，然后agent服务器把agent的状态信息与程序代码序列化(serialized)为串行数据；接着，如果外送的请求成功，系统会将agent的执行动作结束。然后，系统会将串行数据附上相关的系统信息，包括系统名称以及agent的id等，并以比特流(bit stream)通过网络传送至远端工作站。Agent服务器首先提供一个“环境”(context)来管理agent的基本行为：比如“产生”(create)、“复制”(clone)或“分派”(dispatch)agent到远端工作站、“召回”(retract)远端的agent或“暂停”(pause)、“唤醒”(active)agent，以及“移除”(dispose)agent等。

通过这样的机制，agent服务器就可以控制移动agent在入侵检测系统中的活动了。

6 未来的工作

MADIDS模型可以解决普通分布式IDS系统中的网络传输负载过大、核心计算处理瓶颈、网络传输时延等问题。通过智能agent技术，有效地提高IDS系统的整体性能。在MADIDS的未来研究工作中，还需要重点研究如下问题：

(1)如何有效地降低agent的负荷。agent除了执行普通的IDS检测运算以外，还要智能地处理与服务器的数据通信等，因此，需要优化agent的算法以降低对系统的负荷。

(2)agent协商模型有待进一步讨论。

(3)负载平衡处理降低了agent处理的时效性，需要再讨论在负载平衡和处理实效两者之间的折中。

结束语　本文提出了一种基于移动Agent的新型入侵检测系统——MADIDS。MADIDS由事件产生agent、事件分析agent、事件跟踪agent、agent服务器几个部分组成。在agent服务器的控制下，agent智能、自主地分别执行入侵检测任务。文中讨论了MADIDS中的agent协调模型，agent负载平衡，agent重构等几个关键问题，同时也探讨了MADIDS系统未来的工作。综上所述，MADIDS系统可以克服传统分布式IDS系统的不足，是一种新的分布式IDS系统设计思路。

参考文献

1 张云勇. 移动agent及其应用. 清华大学出版社，2002. 1
2 范玉顺，曹军威. 多代理系统理论、方法与应用. 清华大学出版社，斯普林格出版社，2002. 5
3 李旺，吴礼发，胡谷雨. 分布式网络入侵检测系统NetNumen的设计与实现. 软件学报，2002，13(8)
4 唐正军. 网络入侵检测系统的设计与实现. 电子工业出版社，2002. 4
5 韩东海，王超，李群. 入侵检测系统实例剖析. 清华大学出版社，2002. 5
6 Gelernter D，Carriero N. Coordination Languages and their Significance. Communications of the ACM，1992，35(2)
7 Papadopoulos G A，Arbab F. Coordination Models and Languages. Adances in Computers. Academic Press，1998
8 Cybenko G. Dynamic load balancing for distributed memory multiprocessors. Journal of parallel and Distributed Computing，1989，7(2)
9 Willebeek-Lemair M H，Reeves A P. Strategies for dynamic load balancing on highly parallel computers. IEEEE Transactions on Parallel and Distributed Systems，1993，9(4)
10 Jie W. Distributed System Design. CRC Press，2001
11 Sun Microsystems，Inc，Mountain View，California. Network Programming Guide. 1990
12 Stevens W R. Unix Network programming Networking APIs：Sockets and XTI. Prentice Hall PTR
13 Denial of Service Attacks. http：// www. cert. org/tech_tips/denial_of_service. html