第 32 卷 第 20 期 计 算 机 工 程 2006 年 10 月
Vol.32 № 20 Computer Engineering October 2006
·安全技术· 文章编号：1000—3428(2006)20—0175—02 文献标识码：A 中图分类号：TP309

自驱动安全管理策略模型的研究与实现

周 栋，李 伟，李海杰，李光亚
(万达信息股份有限公司，上海 200233)

摘 要：针对目前安全产品各自为政、管理复杂、安全事件与安全信息无法共享，存在着大量的安全信息资源利用率低、难以自动流转等诸多问题，提出和设计了自驱动的安全管理策略模型，围绕该模型，介绍了定义、规范和实现技术。
关键词：安全策略；规则引擎；自驱动；工作流

Research and Implementation of Self-driven Model of Security Management Policy

ZHOU Dong, LI Wei, Li Haijie, LI Guangya
(Wonders Information Co., Ltd., Shanghai 200233)

【Abstract】 To solve many problems in the current security products, such as low integration of these products, the complexity of management, incapable share of safe events and safe information, low utilization of many safe information resource and the difficulty of self-driven, this paper designs a self-driven model of security management policy. According to the model, it introduces its definition, criterion and implementation technologies.
【Key words】 Security policy; Rule engine; Self-driven; Workflow

1 概述

随着国家信息化基础建设的不断建设和完善及全球信息化时代的到来，计算机和网络的使用日益普及，但随之也带来了大量的安全问题。从安全层面上，可以将整个信息系统的安全划分为应用、系统、网络、物理 4 个层面[1]，见表 1。

表 1 信息系统安全的 4 个层面

安全层次	安全威胁	安全需求
应用级	信息泄露，计算机病毒破坏，篡改，假冒，抵赖，盗用	防止信息泄露和破坏，防止收发抵赖，防止非授权使用
系统级	非法访问，黑客入侵，信息截取，欺骗攻击，拒绝服务	隔离，认证，标记，访问控制，安全管理
网络级	非法访问，信息截收，流量分析，数据修改，重放，阻塞，隐信道	隔离，认证，标记，访问控制，安全协议
物理级	窃听、电磁侦收，干扰	物理与环境安全，抗侦收，抗干扰

企业和政府采用的安全产品方面，大致包括：(1)主机监控审计；(2)防火墙、入侵检测、IPS；(3)网络审计(邮件、HTTP、Telnet 等)；(4)数据库审计；(5)反病毒。

在安全产品具体实施的过程中，存在以下问题：

(1)对于中小型企业客户

由于自身比较缺乏技术能力，面对不同的网络安全产品，他们往往手足无措：必须要对每个单项产品在不同的操作界面上进行配置，而且还要考虑到给不同网络安全产品之间的“联动”制定定义和具体举措。很多 IT 技术专家都很难解决的事情，让这些中小企业去做，实在是勉为其难。

(2)对于大企业、政府行业的客户

由于对系统的整体安全性、运行效率、可靠性要求很高，而对价格的敏感度相对偏低，因此在产品的选型上(如防火墙、入侵检测等)往往会采用一些国际知名、性能超群，但功能相对专一的产品。这些安全产品出自不同的公司，产品之间的集成性相对较差，对系统管理人员的要求很高，且工作量大，难以做到联动与综合分析。

凡此种种，当然与我国信息化总体水平有关。其中，缺乏一整套配置简单明了、能够自动执行、可以灵活定制的安全策略也是其中的一个问题。本文提出和设计了一种自驱动的安全管理策略模型，围绕该模型，介绍了定义和规范，对该系统架构的实现进行了简单的介绍。

2 策略模型的定义

传统的安全策略包括物理安全策略、访问控制策略、信息加密策略、网管安全策略等[2]。这些安全策略的实现一般需要运行多个不同的安全产品，使使用者缺乏一个整体把握，因此本文提出自驱动安全管理策略模型来解决这一问题。

自驱动安全管理策略模型是指能够自动根据预定义好的各个策略节点、根据每一节点的执行结果进行自我驱动、自动执行的安全策略，可以跨安全产品实现多种安全产品之间联动的一种安全策略模型。

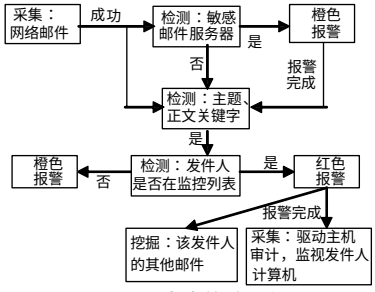

图 1 安全策略示例

一条安全策略由多个不同类型的策略节点构成，根据前个节点的执行结果去自动驱动下一个节点的执行，这就构成

基金项目：国家“863”计划基金资助项目(2003AA148040)
作者简介：周 栋(1973－)，男，工程师，主研方向：构件，中间件，安全技术；李 伟，硕士；李海杰，工程师；李光亚，高工
收稿日期：2005-11-23 **E-mail：**zhoudong@wondersgroup.com

了一条能够跨资源进行自我驱动的执行链，以工作流的方式执行，如图 1 所示。

针对安全领域，将安全策略的节点归纳为：

(1)采集节点：主动采集或接收各类软硬件设备的数据、网络数据包形成规范的、可辨识的安全信息，或根据特定的接口驱动其他安全产品。

(2)检测节点：根据用户自定义的检测规则，对安全信息进行各种检测(如关键字匹配、正规式匹配、逻辑运算等)，产生检测结果，可采用规则引擎、数据库检索、表达式解析等多种方式实现。

(3)报警节点：根据预定义的报警方式进行报警，主流的包括蜂鸣、滚动窗口、弹出窗口、邮件、SNMP、短信等。

(4)挖掘节点：对已经存储的安全信息进行数据挖掘，包括关联、分类、序列、聚类等数据挖掘算法。

从流行的安全模型(PDR[3]模型、P^2DR[4]模型等)来看，其中P^2DR模型包含 4 个主要部分：Policy(策略)，Protection(防护)，Detection(检测)，Response(响应)。防护、检测和响应组成了一个较完整的、动态的安全循环。体现了在安全策略的指导下保证信息系统的安全，自驱动安全管理策略模型把孤立的、分散在各个安全产品的部分有机地融合起来，形成了一整套完整的安全策略，并可以不断扩充以丰富新的功能。

3 策略模型的规范

为了使各策略节点能够协同运行，定义如下规范：

(1)通信规范：考虑到跨平台和跨产品的实际需求，要求在通信方式上支持同步/异步、订阅/发布、广播等多种通信方式；支持 Socket、HTTP、WebService、XML-RPC 等多种通信方式；支持消息的加密、压缩、可靠传递与离线恢复；同时需要使用大数据量和大并发量的要求。

(2)数据规范：数据格式采用 XML 格式以增加其通用性。对安全信息的表示、节点之间的交互均需要定义相应的命令格式。

(3)实施对象规范：定义策略节点作用对象(系统软件、硬件、安全产品等)的表示方式，至少需要定义单个独立对象和上一节点对象这 2 种形式。

(4)时间调度规范：定义策略的执行时间，至少需要定义自动执行、例外不执行 2 种格式。

(5)执行结果/条件规范：定义节点执行结果，用以判断驱动下一个节点，至少需要定义成功/失败 2 种格式。

4 策略模型的设计与实现

自驱动安全策略模型采用分布式 Agent 的方式实现，如图 2 所示，整个系统包括通信平台、策略节点 Agent、管理控制台 3 部分。

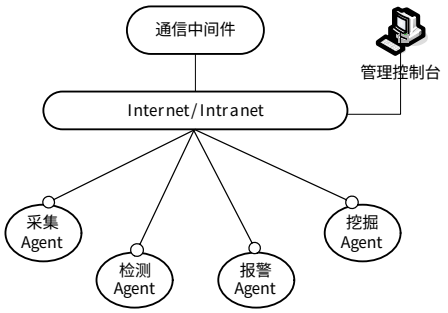

图 2 模型实现架构

(1)通信平台

通信平台采用基于 XML 格式的消息中间件，其由 100%纯 Java 编制，提供 Java 和 Win32 两种 Client API 接口。利用中间件提供的多种通信方式(同步/异步、订阅/发布、广播等)、多种通信协议(SOCKET、HTTP、WebService、XML-RPC 等)、以及对消息的加密、压缩、数字证书认证、可靠传递与离线恢复等特性来满足整个平台的通信要求。

(2)策略节点 Agent

策略节点采用 Java 编制，各类型节点(采集、检测、报警、挖掘)均采用同一框架，在应用层实现各自不同的功能，并实现灵活的扩充(已经实现的功能包括配置采集、网络旁路监控、主机审计、数据库审计、应用审计等)。

1)数据规范(安全信息)。如图 3 所示，采用W^4OAV(Who，Where，When，What，OnWhat，Action，Variable)七元组统一格式来表示安全信息，以达成安全信息的互识别。

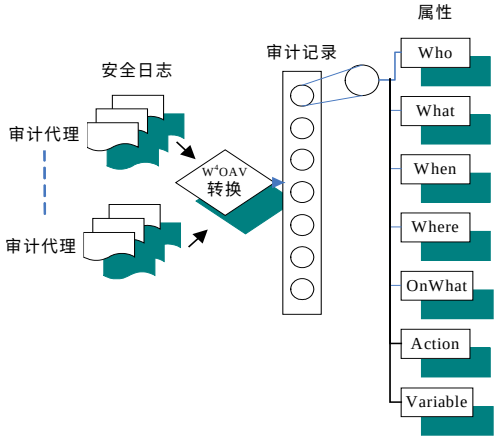

图 3 安全信息七元组

2)数据规范(节点通信)。各节点统一接收通信平台发出的指令，并将执行结果返回通信平台，各节点之间不直接进行通信。节点接收的指令格式为：

```
<CMD>
 <ACTION>命令</ACTION>
 <ACTDATA>命令参数</ACTDATA>
</CMD>
```

实现的命令包括：ADDPOLICY、UPDATEPOLICY、DELPOLICY、STARTPOLICY、STOPPOLICY、RUNPOLICY。

3)实施对象规范。采用 XML 格式来定义策略所实施的对象：

```
<OBJECT>
<ID>对象 ID</ID>
<IP>对象 IP 地址</IP>
<NAME>对象名称</NAME>
<PARAM>对象参数</PARAM>
</OBJECT>
```

4)时间调度规范。时间调度规范采用类似于 Unix 的 crontab 形式的参数表示，这里不再赘述。

5)执行结果/条件规范。节点执行结果由节点返回通信平台，采用 XML 统一格式描述。

```
<RST>
    <RESULT>执行结果</RESULT>
    <DESC>执行结果描述</DESC>
<RST>
```

6)规则引擎。规则检测部分，采用 Jess 规则引擎作为核心，在规则引擎的基础上，前置数据过滤引擎(利用数据库的高效查询)，使规则引擎处理的数据量大为减少，提高了效率，较少了系统资源的占用。

(3)管理控制台

管理控制台为整个模型的前端管理工具，其功能包括规则模板的定义与管理、策略的定义与管理、执行监控、报警监控、查询与统计分析等，不再详述。

(下转第 184 页)