第34卷 第17期 计 算 机 工 程 2008年9月
Vol.34 No.17 Computer Engineering September 2008
·安全技术· 文章编号：1000—3428(2008)17—0173—03 文献标识码：A 中图分类号：TP309

安全事件综合分析系统框架及关键技术

李洪江，周保群，赵 彬
(解放军信息工程大学电子技术学院，郑州 450004)

摘 要：为保证网络安全，布置入侵检测系统、防火墙、防病毒软件等安全产品易造成入侵检测系统的漏报和误报、防火墙的日志信息过于庞大等问题，导致整个系统的安全难以保证。该文提出一种安全事件综合分析处理系统，经过系统的事件收集与预处理、告警压制与聚合、攻击重构和关联结果分析及处理等过程，在一定程度上解决了入侵告警的误报和漏报问题，同时使得管理员更容易获得系统的整体安全状况。对系统中2种核心技术——压制聚合和攻击重构进行了描述。

关键词：日志和告警；系统框架；聚合；攻击重构

Framework and Key Technology for Security Event Synthetically Analysis System

LI Hong-jiang, ZHOU Bao-qun, ZHAO Bin
(Institute of Electronic Technology, PLA Information Engineering University, Zhengzhou 450004)

【Abstract】 It is necessary to install some security equipment or system, such as IDS, firewall, integrality check system etc. to ensure network security. But, in this situation, not only would actual alerts be mixed with false alerts, but the amount of alerts would also be too much to manage. This paper presents a framework for security events, which is mainly composed of following steps: event collection, event preprocess, event condensation, event aggregation, attacks reconstruction and the result analysis. Through this process, the rate of false and miss alert to some extent can be reduced. The aggregation algorithm and attack reconstruction technology are described simply.

【Key words】 log and alert; system framework; aggregation; attack reconstruction

1 概述

传统的信息安全解决方案存在以下几点不足：(1)海量的告警信息，仅一个中型企业的FW每天就能产生超过上亿字节的日志数据；(2)极高的漏报率、尤其是误报率，有90%以上的都是误报，真正有威胁的数据经常被淹没在这些海量的数据当中；(3)告警信息的语义单一可读性较差；(4)传统网络安全信息分析依赖手工操作，工作效率低、成本高、效果差；(5)非集成解决方案，各产品各自为战，管理员很难掌握整个系统的安全态势及作出及时的响应。

为了解决上述问题，本文提出一种全面的、动态的安全事件综合分析系统框架，并对其中的关键技术进行介绍。

2 安全事件综合分析系统框架

安全事件综合分析系统框架如图1所示。

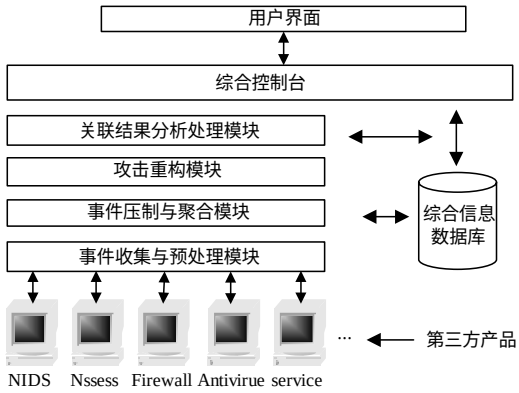

图1 安全事件综合分析系统框架

该系统框架包括以下几个模块：事件收集及预处理模块，综合信息数据库，事件压制聚合模块，攻击重构模块，关联结果分析处理模块，综合控制台，用户界面。

2.1 事件收集与预处理模块

主要由布置在各个网段的可跨平台的代理收集各个安全产品的日志和告警数据及系统的配置信息等，并对它们做出预处理。主要是对入侵告警数据按照IDMEF[1]格式进行扩展性的统一，IDMEF格式形如：

aid:Analyzerid;//产生报警的分析器id
ct:Createtime;//报警产生时间
cl:Classification;//报警类别
src:Sourceidentification;//导致报警的源
dst:Targetidentification;//导致报警的目标

扩展后的格式为：

aid:Analyzerid;//产生报警的分析器id
ct:Createtime;//报警产生时间
cl:Classification;//报警类别
src:Sourceidentification;//导致报警的源
dst:Targetidentification;//导致报警的目标 protocol 告警产生时使//用的协议
severity//告警的严重度
suggestion//告警的解决建议

基金项目：国家部委基金资助项目
作者简介：李洪江(1977－)，男，硕士研究生，主研方向：信息安全；周保群，教授；赵 彬，博士研究生
收稿日期：2007-09-30 **E-mail：**mail112107@sina.com

其中，*src* 和 *dst* 均是结构类型，原来的只包含 IP 和 PORT 2 个域。扩展后，还要包含 MAC 地址和相关的漏洞信息。如一条 Snort 告警可以被对应地解析成：

{05,2007-06-2622:14:22, :1418:11 ,(25.20.186.26,1029,1:0:5E:7F:FF:FA),(25.20.186.56, 139,0:13:D3:D6:C6:5), CVE-2003-0567,UDP,4, Xref=>http://www. eeye.com/html/Research/ Advisories/AD20040226.html Xref=> http://www.security focus. com/bid /9752}

并对处理后的数据进行存储，同时对防病毒、防火墙、漏洞扫描结果等各类日志数据进行必要的解析并存储。如防病毒产品的日志被解析成 6 元组：

{*host_name, dst_ip, dst_mac, accur_time, product_name, product_version*}

漏洞扫描的结果被解析成 8 元组：

{*host_name, host_ip, detect_time, severice, port, protocol, CVE_id, risk_level*}

2.2 综合信息数据库模块

主要负责存储上述的 2 类数据和由管理员定义的关联规则、算法、访问控制策略等。并能保证与各个模块之间进行安全的通信。

2.3 事件压制与聚合模块

聚合技术[2]主要负责把海量的告警和日志数据根据触发事件进行分类、统计等。近而，对数据进行聚合精简，统计事件之间时序关系及发生频率等。

2.4 攻击重构模块

主要负责挖掘入侵告警事件之间，及其与各种日志数据之间，与运行环境的上下文信息之间的深层关系，进而发现事件之间的因果关系，以达到重现各个攻击的整个过程的目的，包括从攻击的源头或攻击的接入点到攻击向各个网络资产的延伸路线。

2.5 关联结果分析处理模块

主要负责对关联结果的验证及对系统整体风险状况的评估和对高风险事件的响应及事件发展趋势的预测等。

2.6 综合控制台

主要负责整个系统运行状况的监控、维护及运行模式的调整，还包括对布置在各个代理端的安全设备和系统运行状况的监控、各个模块的运行状态、数据库的安全状况及其与各个模块之间通信中安全状况的监控。

2.7 用户界面

主要负责与管理员的动态交互，以查看事件的各种统计信息、关联信息、配置信息、策略信息、网络拓扑信息、响应信息等。

3 关键技术之一：压制聚合技术

本节致力于通过一定的算法挖掘出各个告警相应的原始事件，使得告警信息更可读，同时把那些存在着重复和并发关系的告警聚合为一条原始事件信息进行精简。

3.1 相关定义与实例

定义 1 告警的触发事件，就是触发各类安全设备或系统产生某个或某些告警的底层的原始事件。

例如，一个包含一些特殊代码“ “~^$.—*+()[]{}” ”的 ftp 的连接可能触发网络入侵检测系统(NIDS)产生一个 FTP Glob Expansion 告警，那么这个 ftp 连接就是该告警的触发事件。再例如，一个攻击者发起一个端口扫描(port-scan)，相应的防火墙将会产生数以千计的报警，同时各个网络入侵检测系统也会产生一些相应的告警，而这些由这 2 种安全设备所产生的告警都是由这个端口扫描事件所触发的。再比如，一些告警是由网络结构所引起的，在一些特殊情况下，路由不可到达时，将会产生大量的 ICMP Echo 请求，这势必会引起入侵检测系统产生大量的 ICMP Flood 的入侵信息 ;还有一些负载均衡设备，在一些情况下，如果发出的数据包和返回的数据包经过不同的链路设备时，入侵检测系统也会产生入侵告警；等等。而这些事实上都是误告警。触发事件的形式化为：

aid:Analyzerid;//事件 id

ct:starttime;//首次报警产生时间

et:endtime;//末次报警时间

ect:alertcount//相应的告警数量

cl:Classification;//事件类别

src:Sourceidentification;//事件的源

dst:Targetidentification;//事件的目标 protocol 告警产生时使用的 //协议

severity//事件的严重度

下面是本文定义的几个关于分类聚合的函数：

定义 2 告警信息到触发事件的映射函数：*f map*(*T*)， 输入是一个告警信息：*T*={*aid, st, cl, src, dst, protocol, severity*}，输出是一个原始触发事件或事件集合：*E*=*e* 或 *E* ={*e*1, *e*2, *e*3, *e*4,…}，每个事件 *e* 都有相应的属性值{*aid, st, et, ect, cl, src, dst, protocol, severity*}。

定义 3 触发事件验证与聚合函数$f_{aggregate}$()。对于每个触发事件，首先要判断其是否属于一个正常操作；其次考察攻击事件所针对目标是否存在于人们所关注的系统中，比如：如果一个机器上运行着Unix和Apache Web服务器，他收到了一个攻击IIS的告警，那么这个告警可以不用考虑；最后还要对那些纯属同一事件或存在着从属关系的原始触发事件进行聚合，具体的就是它们的各个属性值相等或满足一定的限定条件。比如：删除目录D:\Program Files\Insight3 与删除目录D:\Program Files\Insight3 下的文件ReadMe.txt存在很明显的从属关系，或者事件*ei*与*ej*之间各个属性值满足这样的关系：*ei.st-ej.st* < *r*并且*ei.et-ej.et* < *r* (其中，*r*是用户指定的限定条件)，其他各个属性值都相等，这时就可以将事件*ei*与*ej*聚合成为一个事件。$f_{aggregate}$()输入是一个原始的触发事件*e*={*aid, st, et, ect, cl, src, dst, protocol, severity*}，如果它与已有的事件满足同一或从属关系，则将其合并到对应的事件集合，否则单独成为一个新的事件列。

定义 4 告警的聚类函数：$f_{cluster}$()，输入是告警信息，输出是各个告警对应的原始的触发事件集合。对于输入的各个告警信息，首先判断是否属于经常误报的那些告警，比如IDS系统中的{ 1:1852 -> robots.txt access ; 1:368 - ICMP ping; 1:384 - ICMP ping; 1:366 - ICMP ping; 1:402 - ICMP port unreachable; 1:408 - ICMP reply; 1:480 - ICMP ping speedera; 1:1365 - RM commant attempt (too many false positives)}等，都是IDS经常产生的一些误报。之后再进行具体的聚合运算。具体过程如下：

$f_{cluster}$(T)={//T是一条告警信息
If(T 不属于公认的那些误报)
{Event e=f_{map}(T) //将告警信息映射到原始的触发事件
if(e) //排除原始触发事件是正常操作行为
{if (e.dst)//确认攻击事件的目标资产存在
{$f_{aggregate}$(e)//将原始的告警触发事件进行聚合处理}}}}

每个告警信息除了上述处理，还可以针对关注的焦点不同进行以下的聚合处理。

3.2 基于源 IP 的聚合

通过此项分类可以知道哪些攻击源是最活跃的，如果某个或某些 IP 地址被作为攻击源 IP 地址被多个安全设备的报警报告，那么这个或这些 IP 地址就有很大的可能性是攻击发起的最初源头。

3.3 基于目标 IP 的聚合

通过此项分类可以知道哪些目标受到攻击的频率最高，如果某个或某些 IP 地址经常被作为攻击目标 IP 地址，尤其是一些关键资产是否成为主要的攻击目标。

3.4 基于时间段的聚合

通过此项分类可以知道哪些时间段攻击发起的频率最高，以便发现攻击者惯用的“作案”时间。

3.5 基于告警类型的聚合

通过此项分类可以知道各个类型攻击频率的高低，从而了解系统所面临的各种威胁。

4 关键技术之二：攻击重构技术

现行各类攻击的过程[3]越来越复杂化，多数攻击属于多步的分布式的攻击。作为网络安全系统中必不可少的入侵检测系统，给出的只是孤立的告警信息，而无从知道告警之间的联系，或某个告警信息的之前和之后环节的信息分别是什么。更为严重的是，由于产品的局限性或是攻击者的狡诈致使一些攻击躲过了入侵检测系统的检测。因此，本文在事件收集模块中收集的安全信息还包括防火墙日志信息及防病毒软件日志信息，及一定数量的主机系统日志和网络通信日志。

本节提出的攻击重构技术从思路上部分借鉴了文献[4]中所说明的入侵告警事件关联方法的思想。而该文中提出方法的局限性在于，首先它只是对入侵告警事件进行关联，入侵告警自身误报尤其是漏报的缺点使得关联效果难以保证。其次它提出的关联树形图如图 2 所示，每个节点代表每个告警，每条边代表告警之间的联系，由于告警事件的不确定性及由推理得到的告警之间关系的模糊性，造成这样的树形图很难与事实相符。本文所提出的攻击重构技术中，所采用的信息还包括其他安全设备的告警及日志信息，以及系统的日志信息和网络中的通信日志信息。另外，在文中的建图过程中，以每个主机为节点，以主机间的通信为边，提高了建图的准确性。

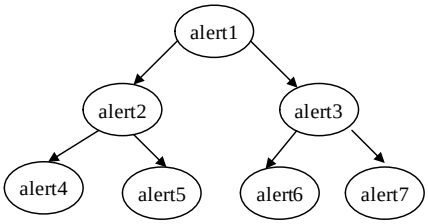

图 2 告警之间关联树形图

首先是对主机间的通信进行建模[5]，*graph G* = (*V*,*E*)，图形的节点$V = H \times T$，H是网络中主机集，T是时间。每条边代表在一定的时间内 2 个终端主机间的通信流，其中包含一个或多个数据包。用一个多维变量$e=\{u, v, ts, te\}$表示每条边。其中，u是通信发起的源主机；v是通信的接收的目的主机；ts, te分别是通信的开始和结束时间；边 e代表从节点(u, ts)到节点(v, te)间的通信。模型中包含的时间是非常重要的，一个主机在被攻陷前后的通信状况或性能是有很大不同的。一个含有蠕虫[6]攻击行为的通信过程模型图如图 3 所示。图 3 显示了假设一个网络遭受了一个蠕虫攻击。左侧的时间早于右侧。图中每个圆圈节点在一定的时间内与各个主机的状态保持一致。在同一个水平线的节点显示了在整个时间段内一个主机状态的变化过程，在同一个铅直线上的节点显示了在同一个时间内不同主机的不同的状态。其中假设经过上一节的告警聚合处理，得到的告警信息都是真实的，在此基础上，根据告警所指向的目标主机就不难标出各个主机在不同时间段里的不同状态。

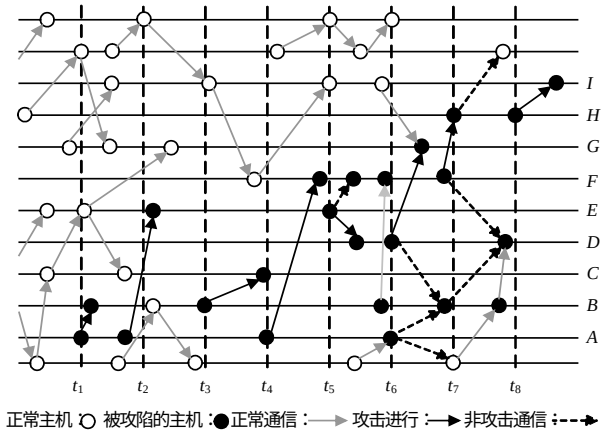

图 3 含有攻击行为的通信过程模型图

图 3 中的每条边代表网络信息流。如果一个信息流没有携带有攻击性的载荷，就称这个边为正常边。作为一个含有攻击性载荷的攻击边，不管它是否攻击成功到目的主机，都用黑色的实线和虚线表示。在一个网段中，攻击可能包含大量的攻击流，只是其中的一小部分可能成功地攻陷一个新主机。如果这样的信息成功地感染了它的目的主机，就用黑实线表示这样的边为攻击进行边。例如：在时间t_6主机D有 2 条边通向主机G和B，只有从D到G的边是攻击进行边，因为主机G是在这次通信被感染的，而主机B在时间t_2之前就被感染了。通过该形式化的模型，可以观察到整个通信过程中攻击是如何进行的，并可以确定最初的攻击源。

为了更直观地看到攻击逐步深入的整个过程，通过从主机通信关系图中提取因果关系边和沿着时间轴突出这些边来形式化攻击的因果关系树形图，即重构攻击的整个过程。

为了在时间概念上与主机通信关系图保持一致，认为在该关系树形图中更高层次的边在时间上发生的更早，图 4 借助标示有时间戳的边来显示攻击的因果关系树，标示有时间戳t_1从攻击源A发出的边位于树形图的最高层。

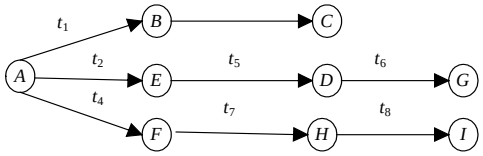

图 4 攻击中的因果关系树形图

鉴于主机通信关系图，攻击重构要识别哪一条边是发生在攻击之前的因果关系边。被感染主机可能继续它们的通信行为，但是，并不是每次通信都能造成攻击的结果，因此，攻击重构必须仔细地推断由被感染主机发出的哪一次通信应该被标记为携带攻击载荷，进而用这些标记好的边重构攻击树。通过追溯攻击树，直到树形图的根结点，就可能确定攻击者的发起点，也或者指明了攻击源的方向，为笔者提供调查的新起点。

（下转第 185 页）