EHFM:一种面向多源网络攻击告警的高效层级化数据过滤方案

杨昕, 李更新, 李挥

引用本文

杨昕, 李更新, 李挥.EHFM:一种面向多源网络攻击告警的高效层级化数据过滤方案[J]. 计算机科学, 2023, 50(2): 324-332.

YANG Xin, LI Gengxin, LI Hui. EHFM:An Efficient Hierarchical Filtering Method for Multi-source Network Malicious Alerts [J]. Computer Science, 2023, 50(2): 324-332.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220800049

EHFM:一种面向多源网络攻击告警的高效层级化数据过滤方案

杨 昕[1] 李更新[1] 李 挥[1,2]
1 北京大学深圳研究生院 广东 深圳 518055
2 鹏城实验室 广东 深圳 518055
(yangxin2016@pku.edu.cn)

摘 要 在复杂网络环境中,态势感知技术根据警报数据实时捕捉多种安全要素及其引起的态势变化,对网络安全进行感知和预测,在安全建设中发挥着重大作用。然而,互联网中海量威胁日志和事件信息带来了极高的分析复杂度,甚至造成了评估和感知技术的误判问题,给安全管理带来了极大挑战。因此,警报事件的过滤起到了重要作用,并且过滤的细粒度、准确性是后续可靠安全态势评估的基础。文中提出了一个面向多源网络攻击告警的层次化数据过滤模型 EHFM,并将其应用于一个安全态势感知系统中。EHFM 包含 5 层过滤器,为多源告警日志设计了统一格式,提出了联合性能熵之差的概念,并结合模糊层次分析等方法,对大量的警报进行统一、精细、定制化的过滤,从而提升安全态势评估算法的准确性、灵活性,解决了网络攻击告警规模过大导致的安全状态误判问题。通过对上述 EHFM 过滤模型和态势感知系统的代码实现,该方案的可行性得到了证明。经过大量实验,结果表明,该方案能够对恶意事件进行精细的分类和过滤,有效避免外界环境因素带来的误判,在大规模网络攻击告警的场景下提升安全态势评估算法的准确性。

关键词:安全分析;层次化警报过滤;多源告警;安全态势感知;模糊层次分析法
中图法分类号 TP393

EHFM:An Efficient Hierarchical Filtering Method for Multi-source Network Malicious Alerts

YANG Xin[1],LI Gengxin[1] and LI Hui[1,2]
1 Peking University Shenzhen Graduate School,Shenzhen,Guangdong 518055,China
2 Peng Cheng Laboratory,Shenzhen,Guangdong 518055,China

Abstract Security situation awareness technology based on the alarm data plays an essential role in system protection. In the complex network environment,situation awareness systems control and predict the network security in time by capturing multiple metrics representing system situations combined with alert data. However,network security detection or protection systems generate massive and diverse alarm logs daily. Such massive threat logs and event information lead to a sharp rise in complexity and even bring some misjudgment problems. Therefore,there is a need for methods that filter the massive warning alerts with fine granularity and high accuracy to provide the basis for building subsequent reliable situation awareness systems. This paper proposes an efficient hierarchical filtering method(EHFM) for multi-source alarm data. EHFM contains five layers of filters,and the proposed hierarchical filtering structure guarantees its scalability and flexibility. Firstly,EHFM designs a unified format for multi-source alarm data to provide unified and customizable filtering. Moreover,the concept of "difference in joint performance entropy" incorporated with the fuzzy analytic hierarchy algorithm is proposed,which guarantees its robustness. These methods improve filtering accuracy by solving the problem of misjudgment caused by excessive alarm scale and external environmental factors. Then,the threat degree of malicious events to the system is classified by considering both the frequency and the impact of alerts. Finally,the classified and filtered alerts are visualized to facilitate the subsequent processing by security managers or software. Based on the proposed EHFM,a security situation awareness system is developed to verify its efficiency. The results of

到稿日期:2022-08-04 返修日期:2022-11-04
基金项目:广东省重点领域研发计划网络信息安全(2019B010137001);国家重点研发计划(2017YFB0803204,2017YFB0803200);深圳市基础研究项目(GXWD20201231165807007-20200807164903001,JCYJ20190808155607340)
This work was supported by the Guangdong Province Research and Development Key Program(2019B010137001),National Key R&D Program of China(2017YFB0803204,2017YFB0803200) and Shenzhen Fundamental Research Programs(GXWD20201231165807007-20200807164903001,JCYJ20190808155607340).
通信作者:李挥(lih64@pkusz.edu.cn)

comprehensive experiments demonstrate that the proposed scheme filters and classifies malicious events in fine granularity and hence improves the accuracy and effectiveness of security situation awareness technology in large-scale alarm scenarios.

Keywords　Security analysis，Hierarchical alarm filtering，Multi-source alerts，Security situation assessment，Fuzzy analytic hierarchy process

1　背景介绍

20 世纪 60 年代以来，网络技术和电子科技经历了从无到有、从简单到复杂的迅猛发展，对现代人类生活和绝大多数领域的发展都产生了巨大影响。随着网络复杂性不断提高，网络犯罪也在不断发展，严重威胁着社会和经济安全。近年来各种各样的安全事件层出不穷，更糟糕的是，网络攻击方式也在不断进化，逐渐由单一模式向复杂高级的可持续威胁攻击（Advanced Persistent Threat，APT）[1] 方向发展。网络犯罪分子利用多种攻击手段，逐渐对网络系统中有价值的资产和关键基础设施发起了广泛的侵略。

网络攻击的复杂多样性与攻击对象的广泛性使得网络安全的防御变得愈加困难。企业或个人通常都要等待安全事件产生才能滞后性地进行防御，难以及时采取有效措施。为了扭转传统防御的被动局面，网络安全态势感知（Network Security Situation Assessment，NSSA）技术应运而生，利用传统入侵检测系统（Intrusion-detection System，IDS）、主机检测系统（Host Intrusion-detection System，HIDS）、web 防火墙等防护系统产生的警报数据，NSSA 可以进行攻击特征提取，从而掌握网络态势，对潜在的攻击进行预判。

然而，在复杂的网络应用环境中，这些安全检测或者防护系统会生成海量的多源网络攻击告警[2]。目前还存在一些困扰，使这些警报难以被态势感知系统和安全运维人员高效利用：1）由于产生警报的软件或设备不同，针对的对象不同，因此其格式比较混乱；2）警报涵盖的范围广、数量多，其中包含大量误报，难以靠手工检查排除误报[3]；3）警报的等级有很多种分类标准，通过网络攻击告警的频次、级别等信息来量化评估警报对系统造成的危害也是比较困难的。

本文针对这些现象，在深入研究现有数据过滤方案的基础上，提出了一种高效层次化精准告警过滤模型（Efficient Hierarchical Filtering Method，EHFM）。该模型不仅能为多源告警日志提供统一的过滤方案，筛除冗余数据，还能避免外部环境因素引起的误判，使得后续的安全态势评估更加精准。由该模型过滤后的恶意攻击告警可以帮助态势感知系统获得更精准的结果，也可直接由上层的其他防御应用进行溯源、学习，并最终能够抵御同类事件的发生。在理论研究方面，本文的创新点主要包括：

（1）为海量多源网络攻击告警提供统一格式的评估、过滤方案，利用统一的五层过滤模型，有效排除影响因子很小的警报，精准地将对系统产生实质性影响的事件过滤出来并进行分类处理。

（2）提出了联合性能熵之差的概念，并结合模糊层次分析法（Fuzzy Analytic Hierarchy Process，FAHP）[4]，平衡单个指标引起的评估结果偏差，排除偶发性意外事件对过滤效果的影响，避免其引起的误报，从而提高算法的鲁棒性。

（3）划分、定义了网络攻击告警的威胁等级和类别，并据此对过滤结果进行分类以及可视化展示，减轻管理员审计与溯源取证的负担，提升运维人员查找警报时的工作效率。

（4）开发基于本文方案的态势感知系统，对本文方案的可用性和有效性进行了验证。

本文第 2 节介绍了相关工作和背景；第 3 节提出了多源告警过滤模型；第 4 节介绍了基于 EHFM 的安全态势感知系统；第 5 节对所提模型、系统进行了实验对比与分析；最后总结全文并展望未来。

2　相关工作和背景知识

2.1　相关工作

基于入侵检测的网络态势评估[5]为多种安全防护系统日志汇总分析问题提供了一种有效解决方案。态势感知主要基于入侵检测进行信息归纳，帮助工作人员更直观地实时掌握系统安全态势，从而采取相应安全防护[6]。然而，如果警报数据量过大并包含无效、误报等噪音数据，安全态势评估模型的评估效果也会受影响。因此，有效地对大量攻击告警数据进行过滤，去掉误报，以合适的粒度提取真实的警报信息至关重要。

在警报过滤技术方面，聚类方法常用于减少需要检查的数据量[7]，以更精准地识别安全事件。Faour 等[8]提出了一种基于传统聚类方法和概率图模型的报警自动过滤体系结构。该结构首先使用聚类算法根据外部机器的相似行进行分组；然后据此确定网络受攻击的情况，使用贝叶斯网络等基于概率学的工具过滤掉误报信息。Chen 等[9]提出了层次化安全态势评估模型，由于该方案切合应用场景，因此受到了广泛关注和引用。He 等[10]提出了一种基于数据融合的警报过滤算法，整合相似度较高的警报再进行过滤，从而降低警报冗余度。Raftopoulos 等[11]概括、总结了一些安全评估指标，针对包括 IDS、黑名单等 4 个日志数据源，提出使用 C4.5 决策树来模拟安全人员对安全日志的分析、决策，并将多数据源日志相互关联。该方案通过对大量不同来源、种类的日志进行关联性分析，提取出抽象的语义和异常事件信息，以此推断异常事件种类。但他们的工作究主要针对告警日志分类，不涉及过滤。

除了单纯的告警日志过滤，现在已经有一些工作将警报过滤应用于安全态势评估中。Yang 等[12]提出了基于连环攻击步骤的安全态势评估算法，结合 k-means 算法筛选，减少安全设备所产生的大量网络攻击告警来提升检测精度，减小了重复警报对态势评估算法的影响。Xi 等[13]提出了一种基于环境属性的安全态势评估方法。但这种威胁过滤方法相对单一，

不涉及对网络攻击告警的评级，存在较为严重的误报问题。

总的来说，目前针对静态分析工具生成的海量告警日志存在误报的问题，已有基于模型检查、语义、规则、切片等不同的解决方案，这些方案各有优缺点，但很多方法的精准性难以摆脱对数据集的依赖，在实际工业环境中很难有理想的效果[14]。而层次化过滤方案不单纯依赖数据集，其结构灵活，可扩展性强，很容易根据场景特征进行调参，从而更能够适应真实场景下的数据过滤。He 等[15]提出了基于多源告警事件关联性的三层过滤方案，这是我们研究工作的先驱和对比方案。该方案首先对多源网络攻击告警根据易受攻击的资源、主机设备和服务的开放端口等特征信息进行第一层过滤；然后捕捉威胁发生时主机设备的性能变化，同时采集真实网络环境下的告警和性能数据，利用性能熵之差对数据进行过滤；最后，以服务器的状态变化为检测基准，进一步划分对应时间片内的警报造成威胁的严重程度，以适应服务器的真实环境。该方案建立了一种数据监控方法，对原始数据进行系统的过滤，对过滤后的告警数据进行威胁等级划分，将其存储起来以供后续使用。但该方案也存在一些缺点，总结如下：

(1)无法有效排除影响因子很小的告警。

(2)第二层过滤无法排除突变点。若是有某个意外突发事件(如网络延迟)引起了单独某个指标超过阈值(Critical Value，CV)，则该时间段所有的数据都会被错误地判断为威胁数据。

(3)威胁程度划分的评估标准单一。仅以告警事件的频次作为依据进行威胁程度划分，没有考虑偶发事件、威胁事件暴露等因素。有些出现频次稍低，但潜在威胁很大的日志有可能会被划分为低威胁告警日志。

2.2 相关概念与定义

定义 1(CVSS) 通用漏洞评分系统(Common Vulnerability Scoring System，CVSS)是安全行业的公开标准，用于评测漏洞的严重程度、所需反应的紧急度和重要度[16]。该系统从被利用难度、暴露的影响力等多角度对漏洞进行量化，给出[0，10]间的评分，其中 10 分对应最严重的漏洞。

本文在实验过程中在系统中部署了部分 CVSS 中的漏洞，然后发起攻击，记录相应的告警日志，将 CVSS 评分作为客观标准，来计算、划分各种类型告警的威胁等级。用户也可以根据需求将其替换成其他开源或内部自定义的评分系统。

定义 2(威胁警报格式) 威胁警报指安全保护软件检测到异常事件并生成的相关警示信息，我们定义其格式如下：$Alert:\{id, srcIP, dstIP, port, service, name, time, level, description\}$。其中，$id$ 是网络攻击告警的标识，$srcIP$ 和 $descIP$ 分别代表源、目的 IP，剩余元素依次为访问的端口、服务、警报名称、告警时间、威胁等级、详细告警信息。这些字段包含了不同种类警报常见的特征，该通用格式可以统一描述大多安全防御软件所生成的网络攻击告警。值得注意的是，由于一些 web 攻击方式，如跨站脚本攻击 XSS、SQL 语句注入等攻击并不会引起剧烈的系统性能变化，因此在第一层识别到这些攻击时，会将其威胁 level 调高，防止其被误过滤为 NORMAL 数据。

定义 3(性能熵) 描述主机和设备性能。对于设备 i 的性能指数 u，其性能熵可以表示为：

$$H_{iu}=|\log_2 P_{iu}| \tag{1}$$

其中，P_{iu} 是第 i 个设备关于指标 u 的标准化参数。

定义 4(性能熵之差) Δt 时间内的性能波动。

$$H_{iu}=\left|\log_2\frac{P_{iu_{\max}}}{V_{iu}}-\log_2\frac{P_{iu_{\min}}}{V_{iu}}\right|=\log_2\frac{P_{iu_{\max}}}{P_{iu_{\min}}} \tag{2}$$

其中，ΔH_{iu} 描述攻击带来的影响；$P_{iu_{\max}}$ 为 Δt 时间范围内第 i 个设备关于性能参数 u 出现的最大值，$P_{iu_{\min}}$ 为出现的最小值；V_{iu} 为第 i 个设备关于性能参数 u 的可用范围最大值。如果 $P_{iu_{\max}}=P_{iu_{\min}}$，那么 $\Delta H_{iu}=0$，表示这段时间内没有攻击造成明显影响。反之，ΔH_{iu} 越大，表示攻击带来的系统变化越大，攻击效果越显著。

最后，介绍模糊层次分析法[4]，其主要用于对系统的多个方面进行量化分析或者计算系统元素的相对重要程度。本文引入该分析法来计算各个性能熵指标之间的权重。FAHP 方法的关键一环是模糊判断矩阵计算，用于后续对应的量化分析。模糊矩阵 $\boldsymbol{A}$ 是一个 $n*n$ 的矩阵，具体表达式如下：

$$\boldsymbol{A}=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{m} \end{bmatrix}$$

其中，n 行 n 列代表系统关注的 n 个指标，每个元素 a_{ij} ($i=1,2,\cdots,n$)代表指标 i 相对于指标 j 的重要程度，且满足：1)$a_{ii}=0.5$；2)$a_{ij}+a_{ji}=1$。a_{ij} 的数值来自对 n 个指标之间相对重要程度的定量描述，采用表 1 所列的标度法进行相关标度。

表 1 重要性度量方法

Table 1 Quantitative description of relative importance

Value	Description
0.9	A 与 B 相比绝对重要
0.8	A 与 B 相比非常重要
0.7	A 与 B 相比比较重要
0.6	A 与 B 相比稍微重要
0.5	A 与 B 同等重要
0.1，0.2，0.3，0.4	B 与 A 相比

最后，根据表 1 计算指标 i 的权重 W_i，计算式如下：

$$W_i=\frac{\sum_{j=1}^{n}a_{ij}+\frac{n}{2}-1}{n(n-1)} \tag{3}$$

3 层次化网络攻击告警过滤模型

层次化多源网络攻击告警过滤模型针对每一个 Δt 时间片内的数据进行过滤。图 1 给出了本文方案的整体结构。本文方案包括 5 层过滤，第一层根据用户需求，提前筛选出特定的条件的数据；第二层根据威胁权重和频率，过滤掉影响因子较小的数据，降低干扰项；第三层利用 FAHP 以及性能熵之差筛除严重告警，并且排除突发环境因素的影响；第四层根据威胁的等级进行排序，对第三层输入的可疑告警日志进行进一步划分；第五层对过滤后的警报信息根据其威胁等级进行分级和图形化展示。

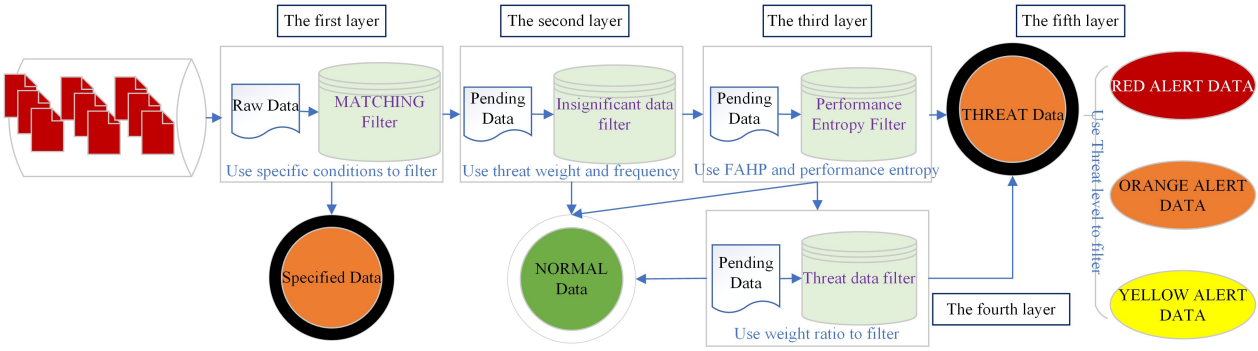

图 1　层次化网络攻击告警过滤模型架构图(电子版为彩图)

Fig. 1　Architecture of hierarchical network attack alarm filtering model

3.1　第一层过滤

当各种设备采集到的数据到达第一层过滤系统时，首先根据其发生时间顺序进行排序，然后根据 Δt 的时间间隔进行分片。如果识别到 XSS、SQL 注入等不会引起系统性能指标剧烈变化的攻击，则将其威胁 level 上调。

安全管理员在第一层过滤器中通过指定过滤条件来筛选想提取的数据，避免它们在其他层过滤器中被过滤掉。过滤时，过滤条件为 $M\{e\mid e\in U\}$，其中 U 为警报信息的所有字段集合，该条件可为空。然后第一层过滤器将会对时间片内的每一个警报 T，检索它的全部字段，当 T 满足 M 内的条件时，此数据将被过滤至 SPECIFIED DATA 数据池中供管理人员后续审计。

例如设定过滤条件 M 为：$\{srcIP:192.168.1.100, service:ssh\}$。这表明管理员想得到 sourceIP 为 192.168.1.100，并且涉及的服务类型为安全外壳 SSH 协议[1]的所有警报。所有符合条件的数据都会被匹配到 SPECIFIED DATA 池中，其余数据进入第二层过滤模型。

3.2　第二层过滤

第二层过滤采用特征统计的方式，根据每一类型事件的威胁级别和该事件出现的频次来判断其影响程度，影响因子较小的事件将被过滤到 NORMAL 数据池中以纯化数据。

特征统计方案是警报过滤研究中一种简单有效的方法，通常选取的网络攻击告警特征有发生时间、警报类型、警报威胁级别、上下文背景、同类型警报的发生频率等。我们对各种特征进行了分析和对比，最终采取威胁级别和发生频率两个特征用于对大规模警报的初次过滤。当服务器受到某种攻击，其威胁级别较小并且发生的频次也不够高时，很难对系统产生严重威胁，那么可以忽略这种攻击的警报。例如：网络上有大量黑客无时无刻对互联网主机的 SSH 等服务进行批次扫描和爆破，服务器的防火墙将会在每次拦截之后生成一个对应的警报。实际上，这种 SSH 口令登录失败对系统的威胁很小，类似的警报便应当被过滤掉。该层过滤器的过滤条件如式(4)所示：

$$\text{Filter if: } alert_i\ level < threshold\ level \ \text{ or } \ \frac{\sum alert_i}{\sum alert_{\Delta t}} < threshold\,freq \tag{4}$$

其中，i 代表某类型的警报，$\dfrac{\sum alert_i}{\sum alert_{\Delta t}}$ 代表该类型的事件在 Δt 时间范围内出现的频率。这里的两个阈值(威胁级别阈值和警报频率阈值)是根据专家经验、相关文献或在部署中通过实验得到的。

3.3　第三层过滤

由第二层进入到第三层过滤器的警报仍包含大量的误报或并未对系统造成威胁和影响的网络攻击告警。针对这种情况，本文利用系统的性能熵之差来度量每一个时间片内的警报对系统引起的变化，同时提出系统所有指标的联合性能熵之差的概念，来更精准地判断该时间片内系统是否因为攻击事件产生了性能变化。结合这两个概念，可以判断系统性能变化是外界环境因素引起的，还是因为攻击者进行了网络攻击。

首先为一个设备选取其性能指标集合，以 CPU，memory，loss，connect，I/O index 为例，其中 CPU 代表 CPU 的消耗率，memory 代表内存消耗率，loss 代表丢包率，connect 代表会话连接数量，I/O index 代表磁盘 IO 的压力。然后根据式(5)，计算第 k 台设备每一个时间片 Δt 内各个指标 i 的性能熵之差 ΔH_{ki}。

$$\Delta H_{ki}=\left|\log_2\frac{P_{ki_{\max}}}{V_{ki}}-\log_2\frac{P_{ki_{\min}}}{V_{ki}}\right|=\log_2\frac{P_{ki_{\max}}}{P_{ki_{\min}}} \tag{5}$$

其中，k 代表第 k 台设备，i 代表 CPU 等性能指标的 index，$P_{ki_{\max}}$ 代表在该时间范围内设备 k 关于指标 i 的最大值，V_{ki} 代表在该事件范围内设备 k 关于性能指标 i 的上限指标值。

得到单位时间 Δt 内各指标的性能熵之差后，将其与预设的阈值进行对比。传统方案中当时间段内的网络攻击告警存在某项指标 $\Delta H_{\text{per}}=\Delta H_{ki}>H_{pi}$，也就是说超过该指标的阈值 H_{pi} 时，该时间片内的警报就会都被过滤掉。但这种方案过滤掉了某些意外事件引起的单个指标异常却不足以对系统构成实质威胁的警报，这产生了大量的误报数据。例如，当无严重威胁事件发生而系统发生被动性网络拥塞时，ΔH_{loss} 会超出对应的阈值，造成大量误报。

为了进一步解决这个问题，本文提出了联合性能熵之差的概念。因为攻击事件对服务器的影响是多方面的，一系列攻击发生时，会引起系统多重性能的变化，而不只是单个性能指标异常。联合性能熵之差需要将多个指标联合起来进行判断，当多指标的性能熵之差联合改变时，才可以确定该时间范围内的事件确实对系统产生了较大的影响。若想综合考虑多种指标的联合性能熵，就需要考虑不同指标对系统整体的不同影响。因此，我们引入 FAHP 模糊层次分析法，首先对

指标进行权重选取，为每个指标分配对应的权重，然后计算系统整体的性能熵之差。根据 2.2 节所述，构建模糊判断矩阵，如式(6)所示：

$$\boldsymbol{A}=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \tag{6}$$

其中，a_{ij} 表示 n 个性能指标之间的相对重要性。模糊判断矩阵构建完成后，本文采用式(7)计算每个指标 i 对应的模糊权重。

$$W_i=\frac{\sum_{j=1}^{n}a_{ij}+\frac{n}{2}-1}{n(n-1)} \tag{7}$$

至此，本文已经求得每个性能指标 i 对应的权重 W_i。那么，第 k 台设备整体的联合性能熵之差如式(8)所示：

$$\Delta H_{\text{total}}=\sum_{i=1}^{n}\Delta H_i\times W_i \tag{8}$$

根据上述指标，可以对计算结果进行对比、过滤。相比性能熵之差判决，本文提出的联合性能熵之差 ΔH_{total} 过滤标准更为严格，也就是说会存在某些数据满足 ΔH_{per} 过滤条件，但是没有达到 ΔH_{total} 过滤条件。这意味着部分指标对应的性能熵之差发生了变化，但是系统的联合性能熵之差并未达到临界值。这可能是受到外界因素的影响，系统个别指标产生了异常。另一种可能是网络攻击导致的个别指标的性能熵之差异常，并未能引起整体联合性能熵之差发生变化，这说明这些攻击的影响相对较小。根据各指标的性能熵之差和联合性能熵之差，可以对数据进行第三次过滤，第三层过滤器的过滤细节如图 2 所示。

(1)如果 $\Delta H_{\text{total}}>H_t$，则 H_t 为预设阈值。该时间片内的警报为恶意警报，即 THREAT DATA。

(2)如果每一个指标的性能熵之差均小于预设阈值，那么对于这一时间段内的所有警报来说，其并没有对系统造成任何实质性的影响，可以将其当作误报，过滤为 NORMAL DATA。

(3) 如果 $\Delta H_{\text{total}}<H_t$，则存在部分指标的性能熵之差超出阈值，那么这批数据就会被送入第四层过滤模型进行进一步的过滤。

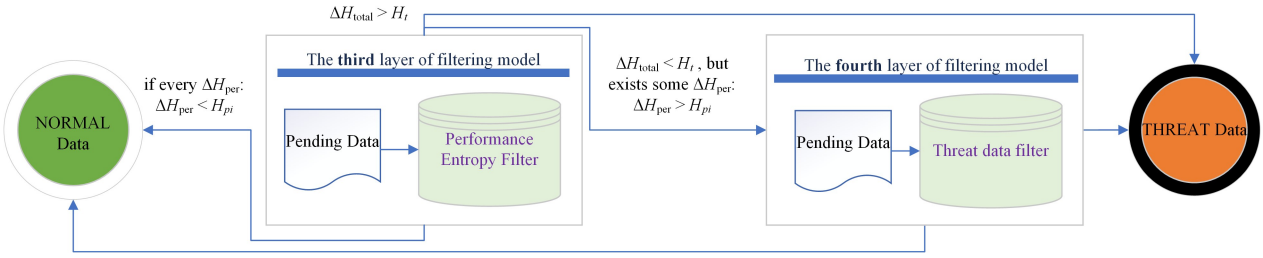

图 2 第三层模型的详细架构图

Fig. 2 Details of the third layer

3.4 第四层过滤

根据第三层过滤器的设计原理，流入第四层的数据存在以下两种解释：1)在 Δt 的时间范围内，确实有某些恶意事件发生，引起了部分指标的性能熵之差发生了变化，但是并未造成严重影响，没有引起全局指标的联合性能熵之差异常，或者不是会引起系统性能指标剧烈变化的攻击；2)部分指标的性能熵之差发生变化是外在因素引起的，而不是事件警报。例如，loss 指标的性能熵之差超出阈值有可能是网络拥塞引起的。这种情况下，该时间分片内的事件应全部标记为 NORMAL DATA。

第四层对上述事件进行进一步的精准分析，筛选出威胁权重更高的告警事件，对以上两种情况进行区分。在这一层，我们对恶意事件的严重程度进行量化评估，统计恶意事件的发生数量、威胁 level 信息，来表述其造成的威胁影响。Δt 时间范围内类型为 k 的事件的威胁程度用 σ_k 表示，过滤原理如下：

$$\sigma_k=C_k\times 10^{L_k}$$
$$\text{Filter if}:\sigma_k>\sigma_v \tag{9}$$

其中，C_k 代表类型为 k 的事件的总数量，L_k 为从 CVSS 获取的关于漏洞 k 对应事件的威胁等级。由于攻击事件造成的影响一般与攻击次数和攻击级别有关，在攻击次数不变的情况下，攻击影响随攻击级别呈指数上升，这种指数上升关系采用底数为 10 的参数来表述，因此得出的关系如式(9)所示。其中 σ_v 为预设阈值参数，当某一漏洞对应的事件满足 $\sigma_k>\sigma_v$ 时，则该事件被过滤为 THREAT DATA；否则不满足该条件的数据为 NORMAL DATA。

3.5 第五层过滤

经过前四层过滤，大量网络攻击告警日志最终被划分为 3 个数据集，分别为 SPECIFIED DATA，NORMAL DATA 和 THREAT DATA。对于 THREAT DATA 数据集，为了方便管理员对威胁事件进行可视化分析并且更直观地了解系统整体威胁状态，第五层根据 CVSS 等级对其进行了进一步划分，并进行了可视化展示。首先获取 THREAT DATA 的 CVSS 数据库评分，其中 8—10 分的威胁事件被定义为红色事件，5—8 分的威胁事件被定义为橙色事件，0—5 分的威胁事件则被定义为黄色事件。分层过后的 THREAT DATA 如图 1 中的最右侧所示，其中红色代表高危级别，橙色代表中等，黄色代表稍弱。

4 融入 EHFM 模型的安全态势感知系统

基于上述警报过滤模型，本节进一步设计并实现了一个安全态势感知系统。网络安全态势感知系统是警报过滤模型的一个常用的应用场景，EHFM 模型过滤后的警报会成为态势评估模型的数据集。感知系统通过态势评估模型的量化评估来展现系统当前的整体安全态势情况，以提供及时的安全

防护。我们将提出的 EHFM 模型应用于该系统中，并模拟工业场景中的安全攻防情况，以更真实地展现该模型对安全防护的作用。本节将从安全态势感知系统相关的设计、拓扑以及实现细节等方面展开介绍。

4.1　框架设计与说明

所提安全态势感知系统的整体架构如图 3 所示。本系统通过采集全网流量数据和安全防护设备日志信息，利用大数据安全分析平台来进行处理和分析，为用户呈现实时的全网攻击态势，为管理者进行安全事件的处置决策提供依据。态势感知系统采用 Endsley 的安全感知模型架构[17]。

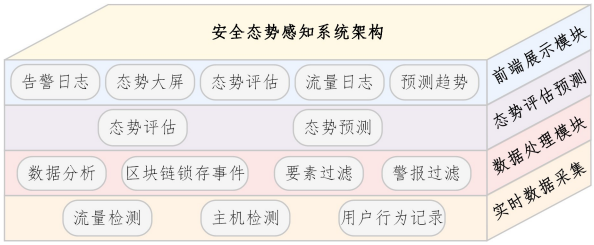

图 3　安全感知系统的整体框架

Fig. 3　Framework of security situational awareness system

本文的态势感知系统主要分为 4 个功能层次。

(1)实时数据采集层。该层包含 3 个子模块：1)流量检测模块，负责采集网络实时出入流量；2)主机检测模块，负责对主机进行安全检测，收集主机的异常告警数据；3)用户行为记录模块，自动记录用户行为，并且锁存到数据库中。数据采集层会对这 3 种类型的数据进行实时采集并向上层传递，以进行数据实时分析。

(2)数据存储与处理层。该层主要对之前收集到的数据进行解析、分类、要素提取并且将其存储起来，然后利用分析算法对流量进行实时检测。

(3)态势评估与预测层。主要负责整合告警日志中所有的异常事件类型，实时对系统的安全性进行评估，并且对系统未来可能的走势进行预测，同时将关键性的安全信息反馈给专业的安全运维人员以进行进一步的策略调控与维护。

(4)前端展示层。将所有威胁信息进行分类整合，进行可视化展示，帮助运维人员清晰直接地了解当前系统的安全情况。

这 4 层之间层层递进，第一层在采集、获取警报信息后，将其交给数据存储与处理层进行解析和存储等相关处理，然后交由态势评估预测层进行汇总评估，最终由前端展示层进行可视化展示。

4.2　模块开发方案与实现

除了 3 个功能层，图 3 还给出了各层的主要功能。从横向分割来看，本系统主要实现了 3 个功能：1)流量检测，对流经路由器的所有流量进行分析；2)主机检测，对服务器上的异常行为或状态进行收集和告警；3)用户行为分析，对用户访问行为进行记录，锁存入数据库。安全态势感知系统采用对应模块化开发，受篇幅限制，本节选取最关键的流量检测与分析和主机检测与分析模块进行介绍，完整代码已上传，请查阅 GitHub[1)]。

实时数据采集层的结构如图 4 所示。其中，日志记录模块由内部服务器对用户流量进行提取解析，处理好的数据将被返回给感知系统，然后由感知系统对用户的访问进行检测，判断用户的访问请求是否合法，并对用户的行为进行分析。

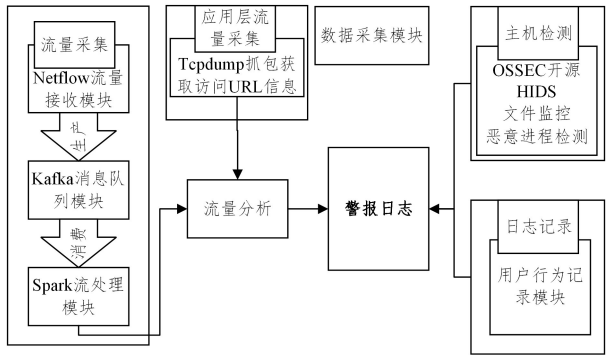

图 4　数据采集模块的结构图

Fig. 4　Architecture of data collection module

在流量采集子模块，分别对经过 Netflow 工具采集处理过的流量信息和由 Tcpdump 捕获的流量信息进行单独分析。利用 Netflow 中的 pmacct 工具对进出服务器的数据包实时采集特征，然后将其记录到数据库中，以方便后续流量分析算法对其进行处理。在得到流量特征信息之后，这些数据被存储到大数据存储系统 kafka 中。

安全感知系统的重点在于系统的数据处理能力，面对高吞吐的流量数据，感知系统要做到低延迟的数据读取、特征提取以及分析等一系列操作，这样才能保证在有攻击事件发生时，系统可以第一时间感知到并进行告警。因此，本文使用了 Spark Streaming 来实时接收数据并进行批处理，以加快系统处理数据的速度，结果将被输送到 kafka 和持久化到数据库中。Spark Streaming 将实时消费 kafka 中的数据，并且通过 Spark Engine 进一步对出入的流量进行特征提取，主要特征为流量的五元组、时间戳和访问频率。

至此完成了流量数据的采集和特征提取，之后进入数据分析阶段。该阶段能识别的恶意攻击事件有 DDoS、主机扫描等。本文选择复现了文献[18]中的基于 ANN 的深度学习算法，经过大量的实验比对之后，该算法对上述几种 IP 攻击方式起到了很好的感知效果。

感知系统的主机检测模块需要将边界服务器作为主要的检测对象，可以考虑采用客户端/服务器模式来运行，即在想要保护的所有的服务器主机上运行代理程序，将边界服务器上的系统记录以及安全防护程序所生成的日志作为数据源，以此来对服务器的运行状态和性能等信息进行分析，若发现可疑事件将即时作出响应。这种模式对于企业或者小型组织用户来说都是相当适用的。

主机检测部分采用开源 HIDS 软件 OSSEC，这本身是一款全面、均衡、成熟且具有很强的系统支持性的主机监控软件。

5　实验与分析

本文在图 5 所示的小型拓扑环境中模拟实际场景，对其

1) https://github.com/xin7777/MIN-SAS_and_IHFM/tree/master

中的服务器进行各种类型的攻击，通过各种检测设备收集多源网络攻击告警并将其传递给感知系统。图 5 中 IP 地址为 192.168.1.31 到 192.168.1.34 的 4 台服务器是内网中常见的应用服务器，提供网络、文件、邮件、数据库等多种服务。IP 地址为 192.168.1.30 的服务器起到防火墙的作用，对来自外网的流量根据指定策略进行隔绝管控。IP 地址为 192.168.1.35 的服务器是部署了安全感知系统的服务器，对所有进出网络的流量进行检测，生成对应的警报。为了使得本实验的过程与效果更加简单明了，本文将 192.168.1.31 到 192.168.1.34 服务器的所有服务映射到 192.168.1.35 服务器上，对该服务器进行大量集中的攻击，来模拟整个生产环境中的威胁情况，放大实验效果。

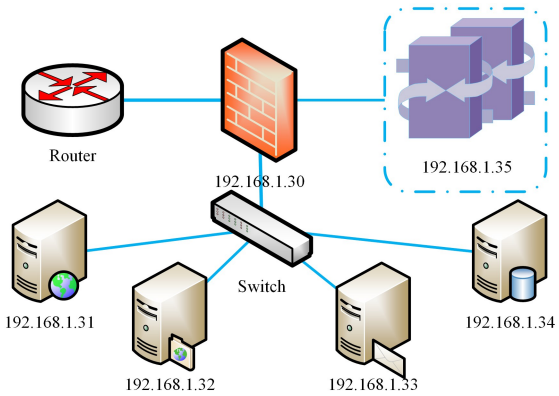

图 5 实验拓扑图

Fig. 5 Experimental topology

实验的每次持续时长为 24 h，在这段时间内，模拟攻击者不间断地对服务器进行各种类型的攻击，例如 SSH 爆破、扫描、DDoS、恶意文件上传等，还模拟了网络拥塞等外部环境突发情况，一共收集到 11 047 条告警日志。

进一步地，Δt 设置为 5 min，以该周期为单位对该范围内的所有事件进行过滤处理。在 24 h 内，我们选择一个时间周期 Δt，在这个时间片中，除了网络攻击还有部分网络拥塞发生，威胁程度较高的攻击相对较少。表 2 列出了该 Δt 时间范围内发生的详细警报信息。实验过程中，2.1 节中的算法[16]被用作对比算法。

表 2 警报类型与数量信息

Table 2 Alarm types and quantity information

警报类型	发生数量	CVSS 等级
FTP 爆破	101	3
Nmap 扫描	49	2
SSH 爆破	155	3
上传大型恶意文件	0	6
DDoS	0	9

本文所有阈值的选取一方面可以根据专家经验、相关文献得到，另一方面可以在具体部署中通过实验得到，可以根据具体应用场景和用户需求进行选择。如果本系统被用于过滤部分数据，向态势感知等后续安全防护系统提供数据输入，那么可以选择相对宽松的过滤阈值，在去除冗余数据和误报数据的前提下，保存更多的数据量以提供给后续系统。如果本系统被用于直接提供分析后的数据给安全运维管理人员，那么可以根据用户的需求选择相对严格的阈值，去除误报和更多的冗余以及影响较小的数据，保留小部分严重危害数据以供工作人员分析。在本实验中，因为后续有态势感知系统进行进一步数据分析，所以阈值选取相对宽松，会保留更多数据。

5.1 过滤过程对比

首先，展示本文算法与对比算法的过滤过程。

第一层过滤：数据送入模型后，在第一层过滤器中由安全管理者指定要提取的特定条件数据，以避免安全管理者想要提取的数据在模型的其他层中被过滤掉。在本实验中暂时不指定任何条件。

第二层过滤：这层过滤会筛选出影响因子小并且发生频率低的事件，根据多次实验，我们选择 $threshold_{\text{level}}=3$，$threshold_{\text{freq}}=20\%$。

第三层过滤：第三层过滤器首先计算进入本层的事件在单位时间内所有指标的最大值与最小值，然后通过式(2)来计算对应的性能熵之差。根据经验判断及实验验证，设 5 个性能熵之差的阈值为 1.04，计算结果如表 3 所列。

表 3 性能熵的差值

Table 3 Difference in performance entropy

Ped	ΔH_{cpu}	ΔH_{memory}	ΔH_{loss}	$\Delta H_{\text{connection}}$	$\Delta H_{\text{i/o}}$
Value	0.44	0.31	1.80	0.53	0.14

考虑 $H_t=CV=1.04$，因此在该场景下，由网络拥塞引起的 ΔH_{loss} 指标的性能熵之差已经超过预定的阈值，因此对比方案会将该周期内的网络攻击告警都过滤为 THREAT DATA。本文方法则需要进一步计算。通过构造模糊矩阵，采用 FAHP 方法可以计算出每个指标对应的权重。进而，根据式(3)求得：$W_{\text{cpu}}=0.25$，$W_{\text{mem}}=0.189$，$W_{\text{loss}}=0.176$，$W_{\text{connection}}=0.199$，$W_{\text{i/o}}=0.186$。最终计算出设备的总性能熵之差为 $\Delta H_{\text{total}}=0.617$，小于阈值 1.04。那么这部分数据虽然存在单个性能熵之差超过阈值，但联合性能熵之差并没有高于阈值。这种情况有可能是网络拥塞等外部环境引起的，因此该时间单位内的事件需要投入第四层进行进一步的解析。

第四层过滤：根据表 2 所列的数据，可计算出各类型告警的 σ 值。通过多次实验，设置 σ_v 为 $2*10^6$。可以发现，该批次所有类型的警报都小于阈值，因此所有数据都归为 NORMAL DATA。

上述 Δt 周期内的实验展示了本文方法与对比方案过滤流程的区别，说明了本文提出的方案能够有效弱化由单个指标超标所引起的误处理的现象，从而避免对特定场景下的警报过滤误报问题，正确判断数据对系统造成的真实威胁程度，过滤效果更加精准。

24 h 内的完整过滤警报数据对比结果如表 4 所列。从整体过滤效果来看，本文方案过滤后的总数据量更低，也就是说，相比对比方案，本文方案更加有效地过滤掉了对系统威胁较小的网络攻击告警。

表 4 告警过滤对比实验

Table 4 Comparison of alarm filtering results

	Baseline	本文方案
过滤前数量	11 047	11 047
过滤后数量	7209	5131
红色警报	1221	1031
橙色警报	2103	1758
黄色警报	3885	2342
误过滤警报	0	0

对比方案在对过滤后的警报进行分级时，只考虑各种告警出现的频率作为划分标准，而本文综合考虑了出现频率和破坏性作为划分标准。表4中，对比方案的红色告警日志数量占过滤后总警报数的16.9%，而本文方案的红色告警日志数量占20.0%。这是因为对比方案忽略破坏性指标后，有部分出现频率较低而潜在威胁严重的数据被降低了告警等级。

对于入侵检测研究来说，除了误报，另一个非常重要的指标是漏报率。通过对过滤为正常日志的数据集进行审计，我们发现两种方案都不存在将高威胁告警过滤为正常告警的情况，即本文方案并没有产生漏报问题。

5.2　多种场景应用实验

本节将本文方案与对比方案在不同场景下进行对比，分别选取了来自4个场景的Δt周期。图6给出了经过两种方案过滤后的警报数量。

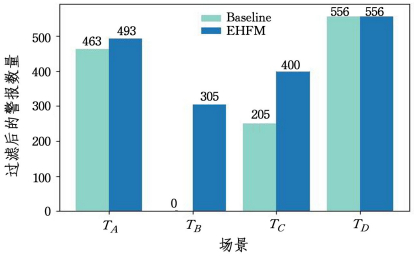

图6　过滤效果对比

Fig. 6　Comparison of filtering results

周期A的场景代表攻击事件造成系统各指标和总的性能熵之差都超过了预设的阈值，并且其中所有告警的威胁程度都大于本文方案第二层的过滤条件，因此这种情况下两者的过滤效果相差无几。

周期B的场景有大量严重威胁告警，此时攻击事件引起的系统单项指标和联合性能熵之差都超过了安全阈值，无论采用哪种方案，这些数据将都会被过滤掉。这种情况下，两种方案的区别在于本文模型的第二层会过滤掉影响因子较小的数据，因此本文模型过滤剩下的数据比对比方案更少。

周期C的场景是网络拥塞且无威胁等级严重的攻击。如图6所示，对比方案将该周期的事件均判为高危数据，属于误判，而本文方案很好地避免了这种情况。

周期D模拟的场景是存在大量某类恶意事件，这些事件造成了部分单个指标的性能熵之差异常，但总性能熵之差并未异常。这种情况下，对比方案会直接将其全部过滤出来，而本文方案在第三层并不会将其过滤出来，而是将其输入第四层，在第四层中通过进一步过滤，相对更严重的报警日志将会被更加精准地过滤出来。

以上实验证明，本文方案相比对比方案能够更好地规避特殊突发事件带来的指标波动，展示出了更强的鲁棒性；对于同样需要被过滤掉的告警日志，本文方案更加精准、灵活，通过调节第四层的过滤阈值，可以提供不同程度的过滤服务。

5.3　性能测试

本节对所提方案与对比方案进行性能消耗实验，分别测试未使用任何警报过滤模型时、使用本文模型时以及使用对比方案时的系统状态和过滤耗时，实验结果分别如表5和图7所示。

表5　性能消耗对比

Table 5　Comparison of system performance consumption

（单位：%）

	CPU	MEM
未使用过滤模型	11.3	23.8
Baseline	11.7	23.9
本文方案	11.6	23.9

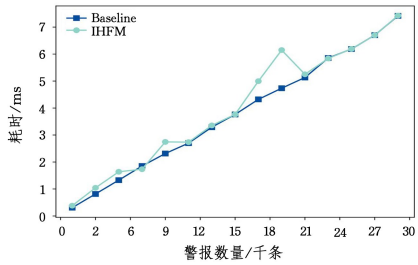

图7　告警过滤耗时对比实验图

Fig. 7　Comparison of filtering time

在资源消耗方面，本文方案与对比方案相差不大，相比完全不运行过滤机制的系统状态，都没有引入明显的开销，不会对系统资源造成过大的负荷。在处理数据耗时方面，本文方案在处理数万规模的警报数据时，耗时大概在毫秒级别。与对比方案在耗时上相差不多。可以观察到，图7中存在个别点本文方案的耗时明显高于对比方案，这是由于在那些数据集中出现了大量的外界环境因素引起告警数据，本文方案花费了一些时间来过滤这些数据。因此，综合来看，本文模型在性能消耗和运行时耗等方面是可以接受的。

5.4　基于EHFM的安全态势感知系统效果测试

接下来将本文模型应用于态势感知系统中，观察使用警报过滤模型优化后的态势评估值曲线与未经警报过滤的态势评估值曲线的区别。在此实验中，告警过滤时间周期设定为2.5 min，态势评估周期设为5 min。

根据图8，该测试过程一共存在3个威胁峰值，告警过滤模型能有效减弱大规模警报中的误报导致的态势评估误差。综合来看，本文提出的警报过滤模型不仅能够减轻管理员的审计、运维等工作的压力，同时对基于警报的态势评估模块也能起到较为明显的作用。管理员通过观察态势评估曲线，能够更加清晰精确地了解当前系统的安全态势。

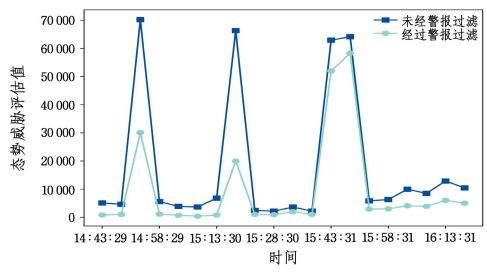

图8　告警过滤耗时对比实验图

Fig. 8　Comparison of filtering time

结束语　针对目前网络安全态势感知研究领域大量警报带来的高计算开销和误判问题，本文在深入研究现有方案的

基础上，提出了高效的层次化多源网络攻击告警过滤方案。该方案为多源告警日志设计了统一格式，提出了联合性能熵之差的概念，结合模糊层次分析、恶意事件评估等方法，有效避免了偶发外界环境因素对算法准确性的影响，提升了方案的实用性、灵活性。进一步地，我们实现了该方案并开发了基于该方案的态势感知系统，并进行了多角度的对比测试。实验结果证明，本文方案在多种场景下都能对系统告警数据进行高效、精准、可定制的过滤，避免了大量外部环境因素带来的误判，能够有效提高态势感知系统判断的效率与准确性。

这只是我们初步的研究工作，受篇幅限制，其中态势感知部分、与其他方法的对比并未展开叙述。未来，我们会对所提方案和态势感知系统进行细化、深入研究与开发，使用更大量的公开网络流量数据集，并将其部署到更大规模的真实网络对抗环境中。深入研究其技术细节，规范参数、阈值选取方案，使所提方案更加具有普适性。另外，对其中其他技术细节（如深度学习算法）进行优化，以提升感知系统对恶意流量的检测能力。

参 考 文 献

[1] LI M, HUANG W, WANG Y, et al. The study of APT attack stage model [C]// Proceedings of IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS). New York: IEEE, 2016: 1-5.

[2] LU X, HAN J, REN Q, et al. Network threat detection based on correlation analysis of multi-platform multi-source alert data [J]. Multimedia Tools and Applications, 2020, 79(45): 33349-33363.

[3] SCARFONE K, SOUPPAYA M, CODY A, et al. Technical guide to information security testing andassessment [J]. NIST Special Publication, 2008, 800(115): 2-25.

[4] VAN LAARHOVEN P J M, PEDRYCZ W. A fuzzy extension of Saaty's priority theory[J]. Fuzzy Sets and Systems, 1983, 11(1/2/3): 229-241.

[5] TANG Z Y, LIU H. Study on Evaluation Method of Network Security Situation under Multi-stage Large-scale Network Attack[J]. Computer Science, 2018, 45(1): 245-248.

[6] BOUTABA R, XIAO J. Network management: State of the art [C]// Proceedings of IFIP World Computer Congress. Boston: Springer, 2002: 127-145.

[7] JULISCH K. Clustering intrusion detection alarms to support root cause analysis[J]. ACM Transactions on Information and System Security (TISSEC), 2003, 6(4): 443-471.

[8] FAOUR A, LERAY P, ETER B. A SOM and Bayesian network architecture for alert filtering in network intrusion detection systems [C]// Proceedings of the 2nd International Conference on Information & Communication Technologies. New York: IEEE, 2006: 3175-3180.

[9] CHEN X Z, ZHENG Q H, GUAN X H, et al. Quantitative hierarchical threat evaluation model for network security[J]. Journal of Software, 2006, 17(4): 885-897.

[10] HE Y, HAN Y J. Research and implementation of an alarm filtering algorithm based on data fusion in NIDS[J]. Science of Western China, 2007, 6(4): 44-47.

[11] RAFTOPOULOS E, EGLI M, DIMITROPOULOS X. Shedding light on log correlation in network forensics analysis [C]// Proceedings of International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment. Berlin: Springer, 2012: 232-241.

[12] YANG X, HUI Z. Intrusion detection alarm filtering technology based on ant colony clustering algorithm [C]// Proceedings of the Sixth International Conference on Intelligent Systems Design and Engineering Applications (ISDEA). New York: IEEE, 2015: 470-473.

[13] XI R, YUN X, ZHANG Y. Quantitative assessment method of cyber threat situation based on environmental attributes [J]. Software Journal, 2015, 26(7): 1638-1649.

[14] AKREMI A. Software security static analysis false alerts handling approaches [J]. International Journal of Advanced Computer Science and Applications, 2021, 12(11): 702-711.

[15] HE X, WANG J, LIU J, et al. Hierarchical filtering method of alerts based on multi-source information correlation analysis [C]// Proceedings of the 27th International Conference on Computer Communication and Networks (ICCCN). New York: IEEE, 2018: 1-6.

[16] Forum of Incident Response and Security Teams, Common Vulnerability Scoring System SIG [EB/OL]. https://www.first.org/cvss/.

[17] WEBB J, AHMAD A, MAYNARD S B, et al. A Situation Awareness Model for Information Security Risk Management [J]. Computers & Security, 2014, 44(2): 1-15.

[18] ABIODUN O I, JANTAN A, OMOLARA A E, et al. State-of-the-art in artificial neural network applications: A survey [J]. Heliyon, 2018, 4(11): 1-42.

YANG Xin, born in 1994, Ph.D candidate. Her main research interests include cyber security, future network architecture, and distributed storage system.

LI Hui, born in 1964, Ph.D, professor, is a member of China Computer Federation. His main research interests include future network architecture, cyberspace security, distributed storage, and blockchain.

（责任编辑：喻藜）