系统工程大作业

科技创新创业项目

申 请 书

项目名称：电网安全态势综合感知系统研发

申请单位：西安交通大学

起止年限：2022 年 03 月～2023 年 03 月

通讯地址：西安市碑林区咸宁西路 28 号

邮政编码：710049

联系电话：029-82668xxx

申请日期：2022 年 03 月

传　　真：029-82668xxx

申 请 人：任泽华

申请学号：3121154002

一、项目可行性调研（包含市场需求、技术发展情况）

1. 市场需求分析

1.1 项目背景

近年来，为建设可靠、安全、经济、高效、环境友好的智能电网，电力系统中引入了大量的信息通信新技术，部署了大量智能电子设备（Intelligence Electronic Device, IED）、远程终端单元（Remote Terminal Unit, RTU）和高级电表量测设施（Advanced Metering Infrastructure, AMI）等智能终端设备。而随着这些信息通信新技术的大量引入和智能终端设备的大量接入，电力系统的网络基础环境也随之变化，网络结构复杂化，网络边界模糊化，安全威胁形态多样化，这给电力系统网络安全防护带来了严峻的挑战。

电力生产关系国计民生，电力网络安全是电力系统安全稳定运行、电力生产可靠的重要保障。随着大数据和云计算时代的到来，网络安全态势的日趋复杂，发电企业网络安全战线延伸，风险点增多，网络黑客或敌对势力针对电力行业的探测、渗透和恶意攻击行为逐步增加，近年来的多起电力系统安全事件也不断警示，信息网络的各种漏洞和脆弱性已经开始影响和威胁到电力系统的安全运行，世界各国的军事力量、黑客组织，都将电力系统信息安全作为重点目标，网络攻击已经成为电力系统安全的重大威胁，发电企业网络安全问题日益凸显。如何保护电力系统的安全稳定运行已经成为国家政治、军事、经济、社会稳定亟待解决的问题之一。

1.2 研究必要性

几起著名的电力系统网络安全事件：

首先是 2010 年 Stuxnet 蠕虫攻击伊朗核电站造成离心机损毁，该种蠕虫病毒通过核电站员工 U 盘摆渡，潜伏在数据采集与监视控制系统长时间收集系统正常运行时的数据，然后利用了 4 个 Zero-day 漏洞和 2 个电子签名认证，来躲过入侵检测系统等信息网络安全监控，借着采用了 PLC Rootkit 技术入侵工控系统控制离心机异常运行，最后借助实施攻击前采集的大量数据伪造离心机的运行数据进行欺骗。

其次是 2015 年 BlackEnergy3 攻击乌克兰电网造成 22 万人失去电力供应，此次攻击共造成 7 座 110KV 和 23 座 35KV 变电站断电长达 3 个小时，并导致 3 个不同区域大约 22 万人失去电力供应。BlackEnergy3 是全球首个导致电力系统瘫痪的网络攻击。Black Energy 的入侵对象是电网控制系统，利用电力系统和 Microsoft Office 的漏洞入

侵电力部门的电脑，之后再通过 VPN 和 ICS（工业控制系统）实施远程管理入侵；在获取了控制系统权限后，伪造和下达断路器断开指令，导致输电网断路；同时通过篡改日志文件、破坏数据存储系统、关闭监控系统以及发动电网客服电话 DDoS 攻击来阻断电网保护和恢复机制。

最后是 2017 年 Staggs 博士团队通过物理连接和控制美国境内无人值守的风力发电机，向网络中其他涡轮机发送命令，禁用或者反复制动急停以造成磨损和破坏；Windworm，利用 Telnet 和 FTP 在可编程自动化控制器间扩散，感染整个风电场的计算机；Windpoison，采用了 ARP 缓存中毒方法，利用控制系统发现和定位网络组件的漏洞，让攻击者可以伪造涡轮机发回的信号，隐瞒遭攻击破坏的事实。他们在美国多个风电场进行测试，撬开风力发电设备的服务器机柜，将通信设备直接物理接入风电控制系统，并实现远程关停电力发电机。

安全事件	网络接入	漏洞利用	指令/数据篡改
伊朗核电站 - Stuxnet	U 盘摆渡	MS08-067 等，控制上位机、PLC 等	伪造控制命令、篡改量测数据欺骗安全监控系统
乌克兰电网 - BlackEnergy3	钓鱼邮件 &VPN	Wincc 高危漏洞，攻击控制系统	伪造控制指令控制电闸开关、阻断电网安全维护系统
美国风电场 -WindShark	黑客物理连接内网	IEC-61400-25 相关漏洞	伪造控制指令操作风力发电机、修改系统状态

通过分析三起著名的电力系统安全事件，我们发现电力系统攻击主要分为“网络接入-漏洞利用-指令/数据篡改”三个阶段。通过网络、存储设备、物理连接等方式，接入电力系统控制网络；利用系统协议、设备和软件的漏洞，入侵部分控制单元和量测设备；根据电力系统业务逻辑和物理规律，构造出符合电力系统规则的控制指令和量测数据，突破电力系统的安全防护体系。

虽然目前工业界和学术界已针对电力系统告警分析与预警开展大量研究，并将各类新型安全技术应用于电力系统安全防御，如网络隔离、身份认证、数据加密、访问控制、入侵检测等，应用多种关联分析方法从告警日志中挖掘潜在的高危攻击行为。但是从不断发生的安全事件中可以发现，现有技术在防御电力系统的信息安全上面临诸多挑战，难以满足电力系统的安全需求，具体包括：

- 海量信息难分析：随着互联网技术的飞速发展，电力系统规模与设备数量大幅增加，攻击日志与告警流量与日俱增，特别是由于告警日志来源的多样性，各种不同的探针设备的生成规则不同，大量的异构数据给现有告警处理分析系统造成了巨大的干扰，时效性和准确性大幅降低。
- 告警关联难挖掘：目前国内大部分电力行业相关企业采用的告警流量监控与分析处理系统均为针对单条告警进行处理，难以应对类似伊朗核电站和乌克兰电网瘫痪等复杂攻击模式。对不同探针设备产生的告警信息、不同来源的告警信息之间潜在的相互关联关系，以及对潜在的链式攻击和借助跳板机等复杂攻击行为难以进行有效地识别与处理。
- 高危告警难检测：虽然　电力系统的入侵检测和隔离系统的安全保障能力已经达到了较高的水平，能够对短时强流量和恶意攻击进行有效地拦截与阻断，但是在伊朗核电站安全事件中，攻击者使用一个月时间对核电站的数据进行监测与记录，这对目前电力系统的安全分析系统提出了全新的挑战，即在海量的低危告警和误报信息中挖掘出长时间持续的扫描探测和高危告警行为。

因此，如果能够从海量告警数据中快速过滤错误告警、对告警之间的关联关系进行挖掘，在中观层面对安全信息进行结构化组织与聚合呈现，将对于改变现有低效模式、提高攻击识别准确度、减少攻击检测时间、辅助安全人员制定防御对策具有重要作用。

1.3 应用前景

随着《中华人民共和国网络安全法》的深入贯彻，网络安全管理工作成为常态化刚性要求。而关键基础设施保护、等级保护 2.0 等相关法律法规的落地，对发电企业的网络安全工作提出了更具体、更规范的要求。集团网络安全管理工作覆盖多种能源形式，涉及海量异构告警数据，目前正在使用的告警分析与处理系统能够对实时告警数据进行初步的规则比对分析与交互式处理，但是随着网络安全态势的日趋复杂，告警数据量呈现急剧上升的趋势，安全分析人员的工作负担加重，现有系统对告警处理的实时性不断下降。

目前电力系统的告警日志来源日趋复杂，由于不同日志采集设备的规则上存在一定的差异性，海量的异构数据对安全分析人员造成了严重的干扰与负担。与此同时，现行的告警分析处理系统对告警过滤的规则设定难以动态更新，误报率较高。态势感

知时空关联综合分析系统通过综合分析不同告警日志来源的数据，从时空关联的角度对低危告警进行批量过滤和高危告警的及时处置，具有较强的应用性与灵活性，极大提升电力系统网络安全防护能力与告警实时处理的能力，并能够及时发现和处置针对电力系统发起的网络攻击行为，有效减少网络攻击带来的经济损失，保障电网安全稳定运行，保障国计民生的安全，具有不可估量的社会效益。

2. 技术发展情况

现代电力系统的可靠运行主要依赖于高度信息化的网络通信与安全防护。如何应对告警流量不断增长带来的分析系统的实时性和告警处理的有效性不断降低的难题，将是新型电力系统面临的重大难题，目前国内外学者广泛提出的解决方案主要有以下三种：(1)利用入侵检测技术识别电力系统通信网络中的异常行为；(2)利用社群发现技术挖掘告警信息间的关联关系；(3)利用时空关联融合分析进行安全态势感知与预警。下面将分别从上述三个方向的国内外研究概况展开叙述。

2.1 入侵检测技术

入侵检测技术是目前检测电力系统等工业控制系统的通信网络中异常行为的主要方法[1]。根据其检测机制，可以将入侵检测技术主要分为误用检测（misuse detection）和行为检测（anomaly-based detection）两大类。

1) 误用检测方法

误用检测也称为基于签名的入侵检测（signature-based detection）。该检测方法根据已知的攻击行为模式设计并构建检测规则库，通过将用户或系统异常行为与规则库中的模式进行比较以检测入侵行为，该种方法对于常见的简单攻击手段效果较好。常用的误用检测技术主要包括基于专家系统的检测方法和基于状态转移分析的检测方法。

(1) 基于专家系统的检测方法

这类方法将入侵的模式转换为 if-then 结构的检测规则。例如，Fovino 等人设计出针对 Modbus 和 DNP3 协议的检测规则，以识别单一报文的攻击类型；同时通过分析报文内容跟踪设备寄存器状态的变化，设计基于设备状态组合的检测规则，以识别出工控网络中的复杂攻击行为[2]。Yang 等人基于对协议和报文的深度解析，构建了 IEC-104 协议的检测规则[3]。

这类检测方法对已知类型的攻击具有良好的检测效果，然而无法检测未知类型的

攻击，同时不能处理数据间的序列相关性。另外，设计规则库时需要考虑规则之间的互斥性，以防止不同规则之间的相互影响产生误报。

(2) 基于状态转移分析的检测方法

这类方法利用有限状态机模型来描述已知攻击导致的系统状态演变模式，使用状态转移图表征不同的入侵或渗透攻击模式。例如，Goldenberg 等人基于对设备 Modbus/TCP 协议报文的深度解析，构建了不同设备流量的确定性有限自动机模型，并通过分析通信过程中的状态转移检测出设备的异常通信行为[4]。Mitchell 和 Chen 设计了一种基于行为规则的入侵检测技术，通过利用电力系统中设备安全运行的行为规则构建表征运行状态好坏的状态转移图，并通过状态转移图检测设备运行状态是否出现异常[5]。

这类检测方法的检测过程只与系统状态变化有关，而与攻击的具体实施过程无关。因此，这类方法可以检测出同类攻击的不同表现形式，对协同攻击和多阶段攻击具有较强的检测能力。然而，攻击模式对应的状态选择依赖于专家经验和人工分析，较难用于复杂的入侵检测场景。

2) 行为检测方法

行为检测的方法假定恶意攻击者的活动异于正常行为，建立系统、网络或用户正常行为的基本模型，并将与之偏离较大的行为判定为入侵行为。这类方法可以检测未知类型的攻击，主要包括基于统计分析的检测方法和基于数据挖掘的检测方法。

(1) 基于统计分析的检测方法

这类方法依据历史数据建立特征变量的统计模型，并通过未来特征变量取值和该模型的偏离程度判定是否存在入侵行为。统计模型包括一元/多元均值与标准偏差模型、Markov 过程模型、时间序列模型等。例如，Fadlullah 等人基于高斯过程的概率模型分析电力系统的通信流量，实现对攻击行为的检测[6]。Hong 等人综合利用主机日志、IEC 61850 协议报文内容等信息，设计了针对智能变电站的综合异常检测系统，可以同时检测针对多个子站的攻击行为，对协同攻击具有较好的防御效果[7]。Zhang 等人研究了一种分布式的入侵检测系统，采用支持向量机和人工免疫系统识别电力系统中的攻击事件[8]。

这类检测方法容易实现，在判决阈值设置合适时可以较好地检测出入侵行为。然而，该类方法的判决阈值较难确定，为了保证低漏报率会导致大量误报。

(2) 基于数据挖掘的检测方法

这类方法借助数据挖掘技术，从大量数据中尽可能地提取隐含在其中的有用信息，并将提取的信息用于进一步的入侵检测。例如，Moghaddass 等人提出了一种分层异常检测框架，基于对海量智能电表的数据分析，检测馈线级别和用户级别的异常事件[9]。Lipcak 等人构建了电力系统用电数据的大数据平台，用于用电数据的异常检测[10]。Marino 等人提出一种基于数据驱动的异常检测方法，使用混合泊松分布建立可应用于大数据集的通信系统流量统计模型，增强了检测结果的可解释性和可展示性[11]。

这类方法需要对大量数据进行分析，对存储和计算资源要求较高。同时，该方法只能在攻击发生之后进行检测，因此缺乏对攻击行为的预警能力。

综上，现有入侵检测方法可通过分析攻击行为的报文内容或通信流量的异常统计特性，实现对恶意攻击设备的高效检测。然而，由于静默监听设备无任何流量特征，伪装响应设备无明显攻击行为。因此，现有入侵检测方法无法从报文内容或流量统计特性上检测出这两类设备行为上的异常，导致检测基本失效。

2.2 社群聚类分析

社群聚类分析技术通过分析告警日志关联图的拓扑结构，将其按照关联程度划分为告警簇，进而深入挖掘分析告警间的关联关系，有效地识别低危误报和异常告警模式。根据社群划分算法的检测机制，可以将其主要分为无重叠型社群划分算法和有重叠型社群划分算法。

1) 无重叠型社群划分算法

此类算法将网络图划分为一系列互相之间没有重叠的区域，每个告警设备仅被划分至一个社群，常用无重叠型社群划分算法主要有以下两个分支：基于图形拓扑的社群划分算法和基于模块密度最大化的社群划分算法。

(1) 基于图形拓扑的社群划分算法

使用图形聚类进行社群划分的两种代表性算法分别是 GN 算法和标签传播算法(label propagation)。GN 算法使用边介数作为表征一条边在网络社群划分中连接两个社群的程度，其核心思想即为不断地从网络中删去边介数最大的边，重新计算所有边介数并循环往复直到网络中最后一条边被删除[12]。标签传播算法则模拟社交网络中的信息传播过程，初始条件下每个节点分别属于不同的社群，每轮迭代的过程中所有节

点倾向于归属于与其多数邻居节点相同的社群，直到每个节点的相同标签邻居节点数目均大于其他标签的邻居节点数目[13]。

此类方法在社群内紧密连接，社群间稀疏连接的网络结构中应用较为广泛，但是电力系统告警日志关联图并不具有此类特征，并且随着告警日志关联图的节点数量急剧增加，算法收敛的条件越来越难以满足，时效性不足。

(2) 基于模块密度最大化的社群划分算法

此类方法主要是通过迭代计算，使得整体社群划分结果的模块度最大，代表算法为鲁汶(Louvain)算法。Louvain 算法首先将网络中所有节点单独作为一个社群，通过遍历迭代所有节点，计算节点归属某一社群后的模块化程度的增益，将每轮迭代划分得到的社团折叠，缩小网络规模，核心是使得整个网络图的模块化程度最高[14]。

此类算法对网络的拓扑结构特征并没有特殊的要求，但是由于其是完全根据网络图的拓扑特征和连通性进行社群划分，如果结合电力系统具体业务场景和专家知识，可能得到较好地社群划分效果。

2) 有重叠型社群划分算法

现实网络中每个告警设备很有可能被多种外部 IP 地址访问或恶意攻击，而如果攻击类型和攻击手段不同，则直观上其应由不同的告警事件所产生，即该告警设别可能同时属于多个社群，因此有重叠型的算法更关注顶点之间的连接关系，而不关心社群间的顶点是否重叠，此类算法主要包括基于多标签传播的社群划分算法和基于拆分介数的社群划分算法。

(1) 基于多标签传播的社群划分算法

此类方法主要是在标签传播算法的基础之上，允许每个节点属于多个社群，常用的有 SLPA 算法和 Copra 算法。SLPA 算法在标签传播 LPA 算法的基础之上将标签的历史传播情况保存下来，通过自定义标签传播与更新规则，来根据不同标签的隶属度计算传播概率[15]。Copra 算法在标签传播算法基础上引入参数控制每个节点最多能够归属的社团个数，通过阈值的判定与随机化的更新迭代保证每个社团中最小的节点数量保持不变[16]。

此类算法在标签传播过程的基础上引入传播规则和参数控制，允许每个节点同时属于多个社群，难点在于如何判断算法的终止时机，电力系统告警日志关联图呈现总体稀疏，局部稠密的特征，如果算法终止条件选择不当很有可能划分出大量尺寸较小

的社群。

(2) 基于划分介数的社群划分算法

此类算法基于无重叠型 GN 算法，引入节点划分介数计算方式，代表算法为 CONGA 算法，该算法维持了 GN 算法通过删除边来进行社团划分的特性，通过比较节点划分介数和边介数的大小来判断是否进行拆分，通过计算最短路径的个数来表征节点连接两个社群的程度[17]。

此类算法对网络的拓扑结构特征要求较高，主要用于不同社群间仅由一个节点相连的情况，对于度数小于 4 的节点将永远不会被拆分，因此并不能够很好地用于电力系统告警日志关联图，但是此类算法通过引入划分介数较好地解释了在什么情况下以及应该如何对节点进行拆分，对有重叠型社群发现算法的发展起到了引领作用。

综上，现有社群划分方法能够对簇内连接紧密和簇间连接稀疏的网络得到很好的社群划分效果，但是目前电力系统的告警日志关联图总体上呈现稀疏的结构，并且不同社群间的连接关系较为复杂，如果仅仅使用拓扑结构进行社群划分基本上无法得到理想的划分效果。

2.3 安全态势感知与预警

安全态势感知与预警技术主要通过对告警信息的关联与聚类分析，分析告警来源的多样性与复杂性，应用常见网络攻击模型，对总体安全态势进行感知与评估，及时对高危攻击事件进行预警。根据安全态势感知与分析预警方法，主要分为告警聚类关联分析方法和日志融合与异常检测方法。

1) 告警聚类关联分析方法

此类方法主要是通过告警信息的统计特征，对潜在的安全风险进行深入分析与挖掘，旨在通过模型规则匹配的方式及时对潜在高危攻击行为进行预警，主要分为告警聚类分析方法和告警关联分析方法。

(1) 告警聚类分析方法

告警聚类将同类型原始告警通过聚类方法，聚合为超级告警，旨在减少告警的数量，去除冗余，便于分析。告警聚类主要分为三类：基于属性相似度、基于专家经验、基于机器学习。基于属性相似度的聚类方法优点在于对相似告警进行聚类时效果好，缺点在于相似度度量和不同项权重分配存在主观因素影响[18]。基于专家经验的聚类方法依赖于丰富的专家知识设计聚类规则，得出的结果可靠度高。基于机器学习的聚

类方法如神经网络算法[19]、K-means 算法[20]等，优点在于不需要先验知识，处理速度快，缺点在于参数的设置和结果的可解释性。

(2) 告警关联分析方法

告警关联通过关联分析低质量告警信息来挖掘深层次的安全风险，旨在攻击意图挖掘和攻击场景重塑。告警关联主要分为：基于前因后果、基于状态转移矩阵、基于场景、基于数据挖掘。基于前因后果的关联方法核心在于借助专家先验知识和历史告警数据，构建一套因果知识网络[21]。其优点在于可以发现攻击意图和新的攻击模式，同时具有过滤误报的功能，缺点在于因果知识网络构建成本高。基于状态转移矩阵的关联方法如隐马尔夫模型[22]，描述了多步攻击的状态转移过程，但时效性还有待提升。基于场景的关联方法如攻击模型描述语言 LAMDBA，具有简单高效的特点，但无法关联未知场景。基于数据挖掘的规则关联方法利用 Apriori 算法[23]、FP-growth 算法[24][25]等挖掘告警的频繁项集和关联规则，缺点是结果的准确性较低。

2) 日志融合与异常检测方法

此类方法主要通过动态时间与告警属性相结合的方式从海量告警数据提取有效信息，挖掘威胁情报，包括基于随机游走的关联规则分析、基于马尔可夫模型的异常检测、基于深度学习的异常预警和基于网络主题挖掘的攻击关联性识别等。朱亮等提出了一套完整的日志挖掘系统，在分析用户搜索点击行为时使用马尔可夫了随机游走的方法，构建用户点击行为关系二部图，预测隐含的关联关系，将搜索相关性提升至71.23%，为日志挖掘和聚类分析提供了良好的思路[25]；张仁斌等在工控网络的异常检测情境下使用混合马尔可夫树的模型，对生产环境中产生的异常情况进行了准确检出[26]；梅御东等则使用了深度学习的方法对系统日志进行分析，对软件系统的异常检测准确率提高了一个档次[27]；Steffen Haas 等使用网络主题特征对网络攻击行为进行聚类分析，对高达 96%的信息成功进行了匹配[28]。

此类方法目前仍然处于研究阶段，在实际的电力系统等工控场景中并没有进行实际的应用，但是其在日志融合和异常检测中所体现的思路对新型电力系统下告警数据融合分析方法的发展起到了一定的指导作用。

二、商业模式分析

电力作为国家关键基础设施之一，在信息系统与电力物理系统深度融合的背景下，我们利用网络态势感知、大数据分析及预测技术，对电网安全攻防状态变化进行全网态势感知，实现智能、联动、快速响应的“主动”防御，实现“知己知彼”的监测目标。我们的系统主要面向企业和政府机构两类用户，针对二者的不同需求，我们设立了不同的商业模式。

1. 面向企业用户

1.1 全量的日志数据采集与解析

全面采集基础架构层、业务数据层和网络设备层的全量日志数据，通过内置的多种解析方式，实现不同格式日志的深度结构化。

1.2 实时的动态视图与监控告警

自定义的动态视图、灵活的监控告警策略以及实时的全局索引搜索，帮助用户全方位洞察日志数据的趋势变化，并可快速应用日志数据。

1.3 智能的日志关联分析与预测

通过搜索处理语言来实现逻辑钻取查询，辅以外部日志数据关联分析，可以综合处理复杂的联动事件，机器学习也可用于预测事件的趋势。

1.4 高效的日志采集部署与响应

线性可扩展的日志采集集群部署方式，可支持传输多达 50 TB 的日志数据。即便是 TB 级的超大数据量，也将于数秒内反馈精准的请求结果

2. 面向政府机构

2.1 设计监控业务

以风险监测与处置为核心，设计平台的业务主线，即“情报收集、深度监测、大数据分析、预警处置和违规调查”。利用安全事件溯源技术追源攻击行为，调查取证安全事件产生的根本原因，核实后进行违规处置。

2.2 建立运营组织

建立安全运营组织架构与协同机制，采用“一级组织，两级应用，前后分工，协同运作”的方式，实现组织人员各司其职，流程横向纵向贯通。

2.3 建设支撑平台

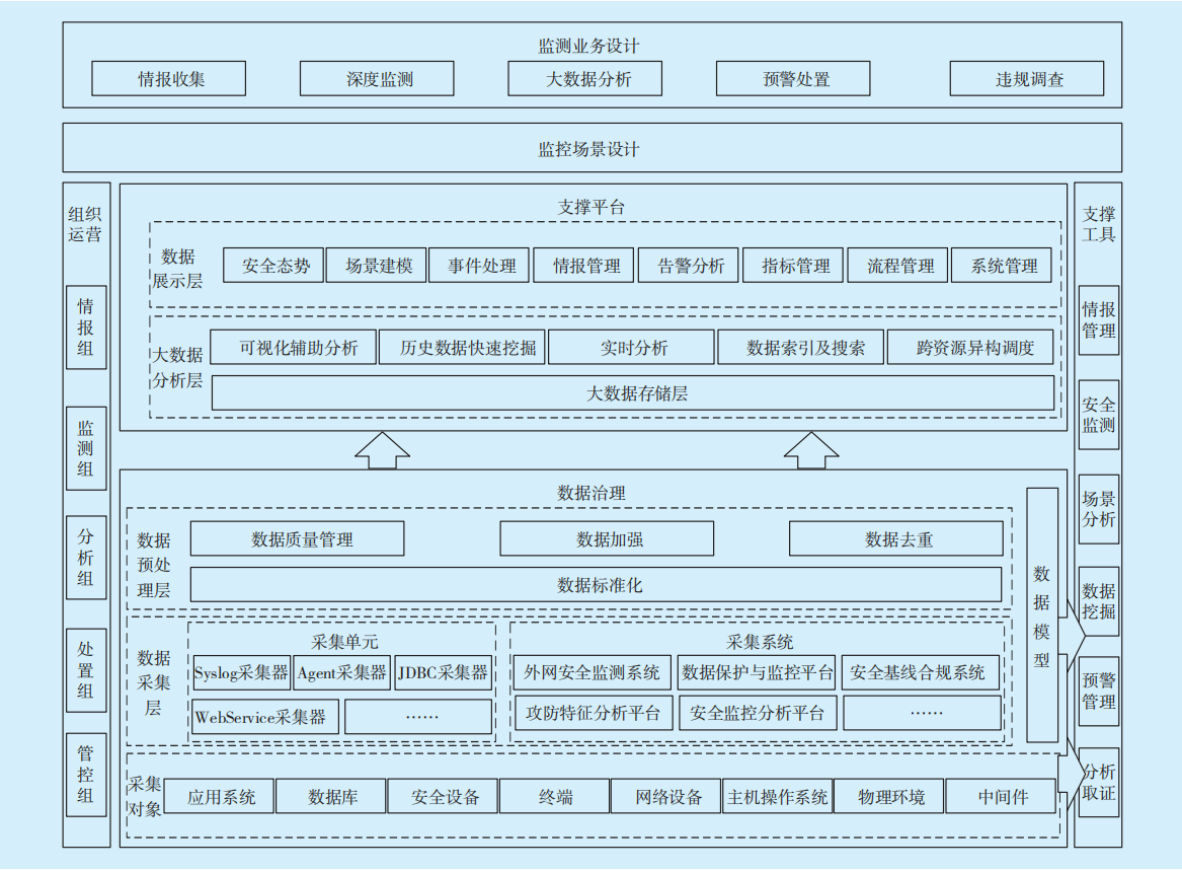
采用网络爬虫、网络捕捉、日志收集、流量镜像等技术全面采集各类安全数据，利用弹性检索技术结合业务场景的威胁检测模型进行安全数据关联分析，满足安全态势全面监控、安全威胁实时预警、安全事件及时处置的业务需求。

2.4 治理安全数据

利用平台规范告警日志分类，实现各安全数据的告警类别标准化，支撑各单位各种设备的规范化接入，为统一态势感知与监测分析提供多源同构的基础数据。

2.5 定制支撑工具

统一情报库管理，逐步挖掘出业务安全事件及其潜在规律，构建起专用化的分析模型技术支撑手段，有力支撑风险监控业务的开展。



三、系统开发目标与内容

1. 系统功能

1.1 安全风险数据分析

基于大数据技术的网络安全态势感知平台一大优势便是对安全威胁数据的有效挖掘，通过对风险数据的有效采集和挖掘，分析判断其是否为威胁情报。先进行数据的预处理，对获得的数据信息来进行特征提取、数据融合、关联分析等加工处理，最终得到基础数据源。然后确认适应性较强的数据挖掘模型，通过对已知攻击方式中的数据报文信息，包括结构特征、数理特征、统计学特征等的分析，选择数据挖掘算法模型的流程、策略与规则。最后便是数据分析，基于数据挖掘模型对预处理后得到的基础数据元做更进一步的分析，判断其中存在的潜在风险，预估存在的安全威胁，完成网络现行态势的感知，为安全防范策略的制定提供数据支持。

1.2 安全态势主动预警

大数据时代最明显的特征便是可实现海量数据的分析处理，并且将其应用到网络安全防范工作，更好的来应对存在的各类潜在威胁。在网络运行过程中产生的大量原始日志信息，涉及到了安全设备、网络设备等，不仅数量巨大，且各类数据之间关联性低，冗余性非常强，无法直接应用网络安全态势感知平台。利用大数据多源信息处理特点，即大数据分布式存储、精确分析以及高效率处理等特征，来完成各类数据的分析处理，作为网络安全态势判断的依据，并且可以形成更加直观的报告，将态势感知结果展示给安全管理人员。

1.3 网络安全主动防御

对以往的网络安全态势感知系统进行分析，可知其通过对网络环境中潜在隐患的分析判断后，仅仅是将结果展示给管理人员，并不能够直接对预测的威胁进行处理。基于大数据技术的网络安全态势感知平台则是进一步对该方面作出了改善，不仅可以处理潜在威胁，还会建立威胁处理特征库，通过机器学习算法对特征库进行动态维护。一旦态势感知系统遭受恶意攻击后，便可以对数据库内的信息特征进行分析匹配，确定最佳应对方法，通过系统入侵检测、防火墙等安全措施的联动，及时处理存在的网络攻击行为，确保系统环境的安全性，以免造成用户的损失。

1.4 功能列表

序号	一级功能	二级功能	功能描述
1	数据源	采集服务器扩容	将采集服务器范围扩展覆盖。
2		采集日志源扩容	采集范围覆盖到生产区、测试区、开放合作区、DMZ区等区域。
3	日志采集及预处理	采集方案优化	持续优化日志采集方案，实现采集脚本快速部署。
4		日志解析	完善数据采集监控功能，实现数据采集监控的可视化。
5		数据监控	解析模板丰富，性能调优。
6	行为分析	账户关联	建立账户关联体系，实现堡垒机与多个系统的账户关联，实现内网账户体系的打通。
7		账户威胁等级评估	建立账户威胁等级评估模型，结合威胁等级、业务重要度、账户权限和破坏性等对账户的威胁程度进行评估。
8		行为异常检测	采用多种方法发现用户异常行为，如基于个体行为特征偏离度的异常行为检测、基于群体聚合状态分离的异常行为检测、基于频繁模式的异常行为片段检测，进行用户异常行为检测，并进行报警；
9		用户画像分析	基于新增数据源和数据种类丰富用户画像维度，组合业务、时间、地址、角色、对象等多种维度丰富用户画像，从而丰富用户行为基线。
10		用户行为轨迹分析	将用户的所有行为根据时序关系进行重构，深化用户行为动作的分类描述。
11		关键业务审计分析	实现业务关键行为和事件审计分析，如脚本执行审计，便于业务相关人员快速对业务情况进行了解。
12		用户行为基线细化	敏感操作基线细化，后续可支持针对同一账户，在不同服务器上有不同的敏感等级。
13	可视化展示	账户行为威胁可视化	在账户威胁等级评估的基础上，对用户提供排名和威胁可视化展示。

14		用户群体画像可视化	在用户画像分析的基础上，对不同用户群的画像数据进行可视化展示。
15		用户个体画像可视化	在用户画像分析的基础上，对用户个体的画像数据进行可视化展示。
16		用户行为轨迹可视化	在用户行为轨迹分析的基础上，结合历史数据给出 5W1H 行为链，并提供可视化展示。
17		关键业务审计可视化	对关键业务审计分析的基础上，将分析结果进行可视化展示。
18		安全审计报告定制化	可根据业务需要定制不同周期的数据报告，包括周报、月报等。

2. 系统性能

指标名称	指标要求	备注
析节点数	10000 至 100000 之间	
告警日志总量	每月 1000000 条以上	
日志更新速度	实时，与 IDS 系统相差不超过 1 秒	
图构建速度	10 秒-1 周的时间间隔	
子图挖掘速度	1 秒以内	
模式匹配速度	1 秒以内	
管理分析准确率	98%以上	

3. 设备信息

3.1 服务器需求

用途	CPU	内存	物理磁盘	网卡	数量
数据接收及预处理服务器	2*8C	256G	4*600G/raid1	10000M	2
数据分析服务器	4*8C	512G	2*480SSD/raid1 12*4T SATA/raid5	10000M	3
数据库服务器	8C	64G	600G	10000M	2
应用服务器	4C	32G	600G	10000M	2

3.2 IP 需求

用途	IP 范围	数量	备注
物理机使用	10.20.14.3~10.20.14.8	6	15 个 IP 可互通，且能够访问 kafka 集群
虚拟 IP	10.20.15.11~10.20.15.19	9	

4. 总体成本

预算支出科目	金额（万元）	备注
1. 研究人员薪酬	60	
2. 燃料、材料和动力费	50	
3. 设备折旧及使用维护费	25	
4. 软件等无形资产摊销费	15	
5. 中间产品试制费	30	
6. 成果论证及评审费	5	
7. 外委试验费	5	
8. 差旅费	5	
9. 项目管理费	5	
合计	200	

四、项目总体设计方案

针对海量的告警数据，我们使用 Louvain 社区发现算法，将其划分为多个内部信息相似的告警簇，大大降低了需要处理的业务数量。对于已有的各种告警簇，我们使用 G-tries 网络主题挖掘算法搜寻其中的频繁子图模式，提取其拓扑信息用于后续的分类。接下来我们结合关键顶点 IP、告警类型等信息对告警簇进行关联融合分析，找到了其中存在的常见业务模式与高危模态。最后基于上述工作成果，我们开发并部署了辅助安全分析的网页系统，将算法分析的结果进行了快速清晰的可视化呈现，可以将原始告警数量降低 1~2 个数量级。

接下来我们根据电网通信子网的真实告警数据，对告警日志进行了关联融合，构建了高效的聚类模型，识别出了特定的攻击关联规则。同时基于带告警类型的攻击图模型，使用先进的图模式匹配算法，找到了其中存在的常见模式与高危模式。开发的安全辅助系统实现了宏观安全态势的时空关联协同感知，完成了高危报警的快速关联挖掘与定位，有助于对海量信息中的误报进行批量的可解释性过滤。

不同的网络与应用场景所对应的方法多种多样。我们当前所面临的大型电力系统网络具有海量告警数据和多源异构的警报信息，且对于安全系统的实时性、稳定性要求很高。所以我们选择了最适于当前场景的基于网络主题的场景识别方法[1]。如图 4-1 所示，（1）首先使用社区发现算法将告警网络分隔成多个内部相近的告警簇；（2）再使用网络主题挖掘算法对不同的簇进行子图模式挖掘；（3）最后根据子图模式信息与顶点 IP、攻击类型等辅助信息对告警簇进行分类，找到常见模式与危险模式，达到感知预警的目的。

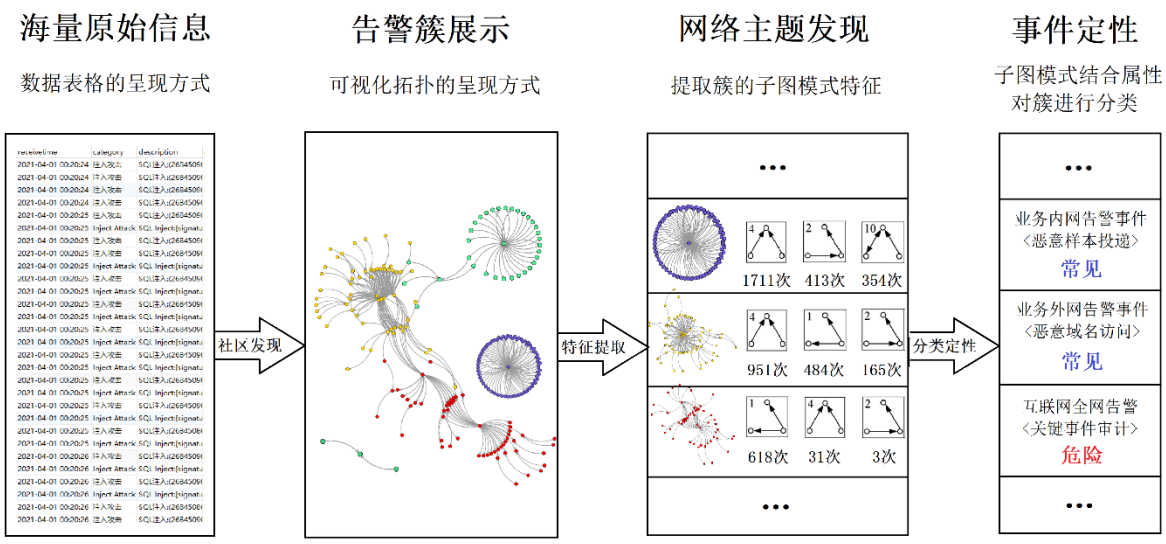


图 4-1 告警信息关联挖掘主要步骤

1. 告警信息聚类分析

1.1 模块化程度与增益度的定义

我们定义网络的模块化程度值为 Q，其计算方法见式 5-1~5-3。其中 m 代表整个网络所有边的权重之和，$\delta(c_i, c_j)$ 是一个判断函数，如果 i，j 属于同一个社区则函数值为 1，否则为 0。这个变量描述了社区内边的权重总和与随机图中权重总和的差别，Q 值越大说明该网络中社区的聚集程度越高，社区中顶点的关系越密切。

$$Q = \frac{1}{2m}\sum_{i,j}[W_{ij} - \frac{k_i k_j}{2m}]\ \delta(c_i, c_j) \tag{5-1}$$

$$m = \frac{1}{2}\sum_{i,j} W_{ij} \tag{5-2}$$

$$\delta(u,v) = \begin{cases} 1\ when\ u = v \\ 0\ else \end{cases} \tag{5-3}$$

式中：Q——网络的模块化程度；m——网络中所有边的权重之和；W_{ij}——顶点 i 到顶点 j 边的权重；k_i——与顶点 i 相邻的所有边的权重之和；k_j——与顶点 j 相邻的所有边的权重之和；c_i, c_j——顶点 i 和顶点 j 所属的社区。

上式中 $k_i k_j / 2m$ 的含义为：网络中顶点 i 和顶点 j 在随即图上相连的期望。这是因为在构建随机图时，每个顶点的出入度保持不变，原始顶点相临边的权值之和除以全部边的权值之和即为该点和其他点相连的概率。由于边的两端有两个点，所以两点相连的概率为二者相邻边权值之和的乘积除以所有边的权值之和。为了更清楚地了解模块化程度的意义，我们对式 5-1 进行了拆分化简，得到了下面的式 5-4。从这个化简式我们可以很清晰地看出：当社区内边的权值之和越大、社区外和社区相连边的权值和越小时，整个网络的模块化程度越高。

$$Q = [\frac{1}{2m}\sum_{i,j} W_{ij} - \frac{\sum_i k_i \sum_j k_j}{2m}]\ \delta(c_i, c_j) \tag{5-4}$$

$$= \frac{1}{2m}\sum_c [\sum_{in} - \frac{(\sum_{tot})^2}{2m}] = \sum_c [\frac{\sum_{in}}{2m} - (\frac{\sum_{tot}}{2m})^2]$$

式中：$\sum_c$ ——对网络中所有的社区进行加和；$\sum_{in}$ ——对社区内部的边的权重加和；

$\sum_{tot}$——对社区外部与社区相连边的权重加和。

在定义了模块化程度 Q 之后，我们接着定义模块化程度增益 ΔQ。顾名思义，ΔQ 就是模块化程度 Q 在算法操作前后的变化，有了这个值我们就可以通过算法迭代，找到模块化程度增加最快的方向，使得整个网络的模块化程度很快得到提高。ΔQ 的计算方法如式 5-5 所示，其中变量的定义和上文相同。从式中我们可以很明显地看出增益度的物理意义：减号前方括号中的内容为将顶点 i 加入某个社区后该社区的模块化程度；减号后方括号中的内容为顶点 i 和它即将加入的社区二者各自模块化程度之和。

$$\Delta Q=[\frac{\sum_{in}+k_{i,in}}{2m}-(\frac{\sum_{tot}+k_i}{2m})^2]-[\frac{\sum_{in}}{2m}-(\frac{\sum_{tot}}{2m})^2-(\frac{k_i}{2m})^2] \quad (5\text{-}5)$$

式中：ΔQ——模块化程度增益；$k_{i,in}$——把顶点 i 加入某个社区以后的整个社区内部顶点之间边权重之和。

上式为增益度的定义式，虽然能够很明显地看出它的物理含义，但却不方便计算，所以我们对式 5-5 进行了进一步的化简得到了式 5-6 所示的简化形式。使用该形式可以很方便地计算模块化程度的增益量，便于进行后续操作。

$$\Delta Q=[\frac{k_{i,in}}{2m}-\frac{\sum_{tot}k_i}{2m^2}]=\frac{1}{2m}(k_{i,in}-\frac{\sum_{tot}k_i}{m}) \quad (5\text{-}6)$$

1.2 Louvain（鲁汶）算法简介

Louvain[5]（译作鲁汶）算法是一种模块化的社区发现算法，它在可以在短时间内获得效果很好的社区聚类结果，并且支持层次结构的发现。Louvain 算法的核心是使得整个网络图的模块化程度最高，下面我们基于这一点对该算法进行简要说明。

算法 1（Louvain）该网络社区发现算法主要分为两步迭代设计：第一步为迭代合并，发现初步的社区识别结果；第二步在第一步结果的基础上对社区进行折叠，再次进行社区发现。通过不断迭代最终获得高效的层次化社区发现结果。

第一步：算法扫描待处理网络的所有顶点，将每个顶点单独作为一个社区，此时每个社区内部权重之和为 0。接着找到每个顶点的邻居顶点，计算把这个顶点纳入邻居顶点所在的社区后模块化程度的增益，将其加入增益最大的社区。不断进行这个过程，将顶点和社区进行合并，直到所有的社区都不再发生变化，获得第一层的社区发现结果。针对这个初步结果，对每一个社区进行折叠，即把一个社区内的所有顶点合并为一个顶点。折叠后每个社区内部边的权重之和变成该顶点自环边的权重，社区之间连接边的权重之和变成顶点之间新临边的权重。

第二步：与第一步起始的过程类似，不同的是此时每个顶点内部自环的权重值不

全为 0。对第一步获得的折叠后网络图进行迭代计算，将模块化程度增益最大的社区与顶点合并，直到所有社区都不发生变化，获得第二层的社区发现结果。同样地，对第二层结果也进行折叠，获得点数更少的网络图。

图 4-2 是一个简单的使用 Louvain 算法进行网络社区发现的示例，通过两步算法的迭代，最终把网络分成了两个大型社区。

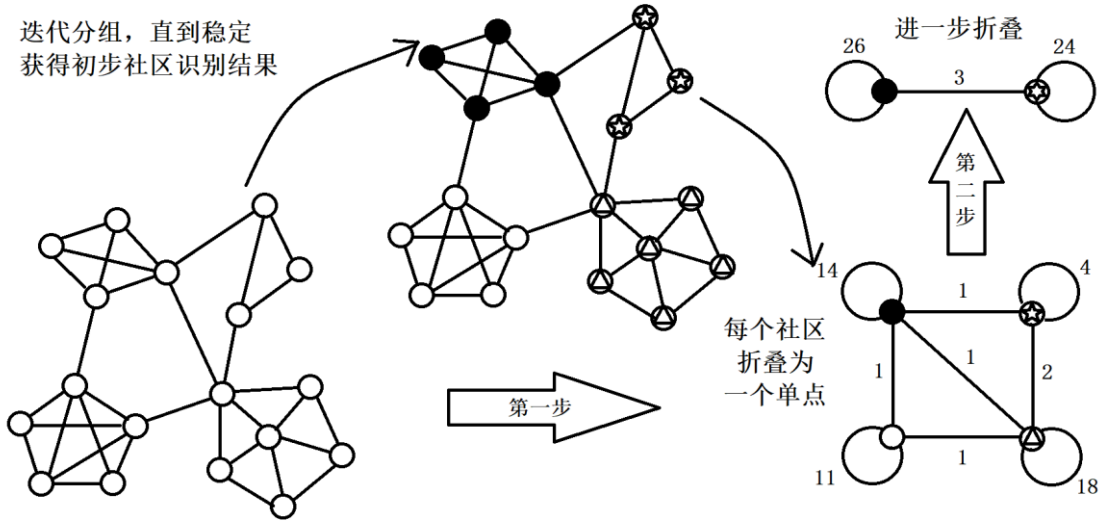
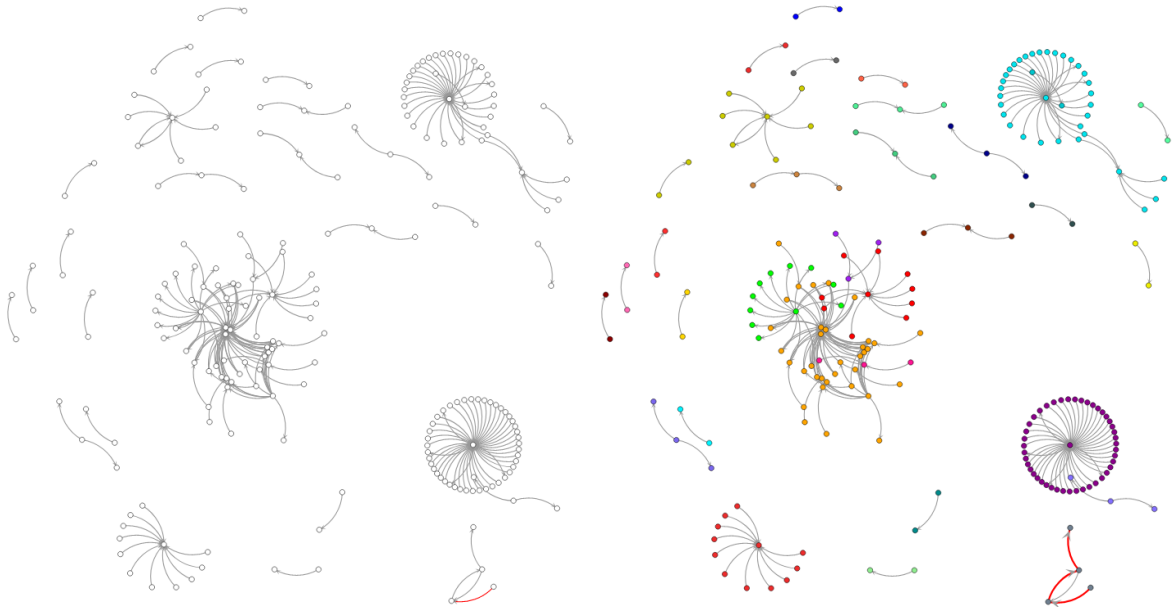


图 4-2 Louvain 算法流程示意图



(a) 聚类前的网络　　(b) 聚类后的网络

图 4-3 告警信息聚类结果可视化

在实际应用中，当网络规模较大且边权重值差别不大（或为无权图）时，分离的小社区有很大概率被合并为同一个社区。这是因为顶点之间权重本身很小时它们连接到任意一个社区的概率都非常小，此时只要两个小社区之间存在一条边都一定会被合

并。而由于我们的网络通常都是上百个顶点和边且边缘无权重，这种问题会更加明显。所以我们采用了图萃取的方法，先给每条边设置权重，通过过滤的方法去掉那些不同社区之间连接较弱的边。通过这种方法我们可以显著提高社区发现的质量。

1.3 告警信息聚类结果

我们使用 Louvain 算法对已有告警数据进行了测试。如图 4-3（a）所示，我们将告警设备作为网络图的顶点，将告警信息作为网络中的边，将其中某一个小时内的告警数据进行了可视化。在这一个小时的时间内，告警事件一共发生了 228 次，其中涉及到的设备有 217 个。也就是说该网络图共有 217 个顶点和 228 条边，对社区发现算法来说是一个比较大的网络。

应用 Louvain 算法发现社区的结果如图 4-3（b），在图中属于同一个社区的顶点被染上了相同的颜色，我们一共将这个时段内的告警分成了 33 个簇。从可视化拓扑中我们可以看出告警信息本身就有许多独立的区域，它们不与其他部分相连。很明显地，我们的算法把它们划分成了独立的社区。

我们在系统中加入了时钟来模拟实时环境。默认告警内容每 10 分钟刷新一次，把前一小时内的所有告警进行绘制。系统也支持修改刷新间隔与窗口时间，在实时性允许的情况下，系统最小支持每半分钟刷新一次；在前端缓存允许的情况下，系统最大支持 2~3 天的数据同时进行绘制。如图 4-4 显示，我们将同一个告警簇内的节点染上了相同的颜色。同时系统还支持点击查看告警详细信息。

我们的系统还支持按照簇节点数量、告警厂商的筛选。通过拓扑图上方的筛选框，我们可以指定仅展示点数在指定范围内的告警簇，也可以仅展示指定厂商设备产生的告警信息，对某一设备的告警按不同类型染上不同的颜色。在右上角可以选择分析模式——实时刷新展示或固定当前内容进行详细分析处理。

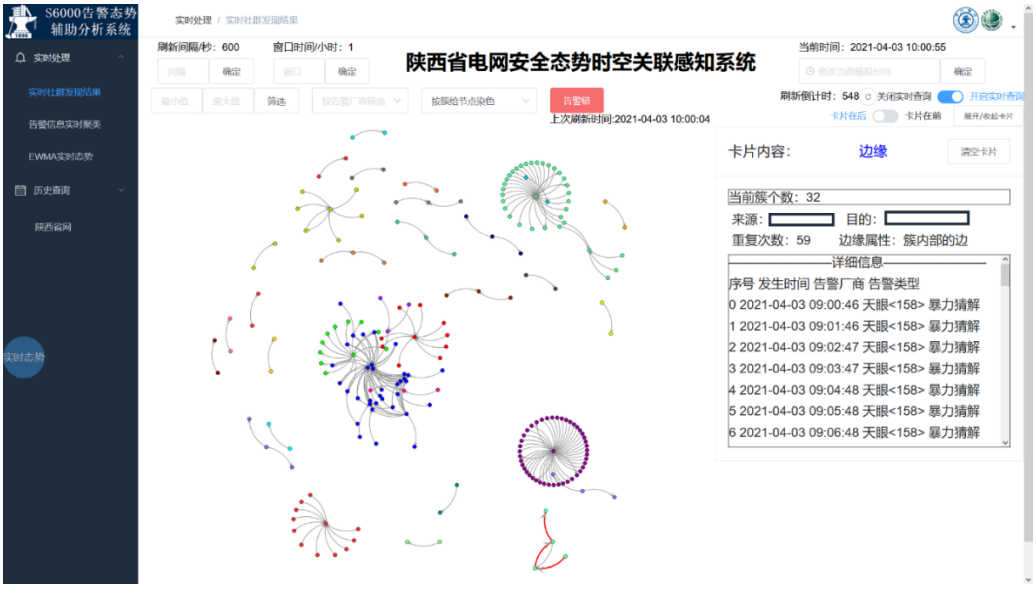


图 4-4 告警簇与告警链可视化界面

2. 告警簇频繁子图挖掘

2.1 G-trie 数据结构的定义

G-trie 是一个能够存储图拓扑信息的多路树[3]，即图字典树。它的名字来源于图（graph）和字典树（tries）的组合。使用 G-trie 数据结构进行网络主题发现的算法就被称作 G-tries 算法。G-trie 中的每个节点都存储了一个子图的结构信息，用于构建整个网络的子图模型。为了避免分歧，在上文介绍网络图涉及到点时，我们一律采用“顶点（vertex）”来进行描述；而在下文中，我们将使用“节点（node）”来指代 G-trie 树的节点，而“顶点”仍用来描述 G-trie 存储的实际网络图中的点。

在进行主题挖掘时，我们需要确定被查找的子图集合并使用其构建 G-trie。再遍历需要进行主题挖掘的网络图，找到其所包含的所有子图，确定 G-trie 中该子图对应的节点，将该节点的频次增加。由于子图之间存在明显的包含派生关系，就如频繁项集挖掘的 FP-growth 算法一样，G-tries 算法也使用了字典树的结构，对子图模式中相同的部分进行了复用。在需要查找某个子图模式出现的次数时只需查找构建的 G-trie 即可，不需要再次对整个网络图进行复查，大大节省了搜索时间。

图 4-5 给出了一个将五种大小、结构不同的子图插入到 G-trie 的步骤，每个子图插入的过程都使用灰色标出，使用黑色标示当前节点对应的子图顶点。由于 G-trie 是一个扩展式的结构，所以树根是一个存储了空子图的空节点。子图节点数越多，其所对应的位置也就越靠下。每个 G-trie 节点都包含了一个新加入的子图顶点的信息以及该节点与其先祖节点之间的关系，这意味着它们是自己所存储子图的构建路径所对应的最后一个节点，即从根节点到这个节点的整条路径定义了整个子图的构建过程。

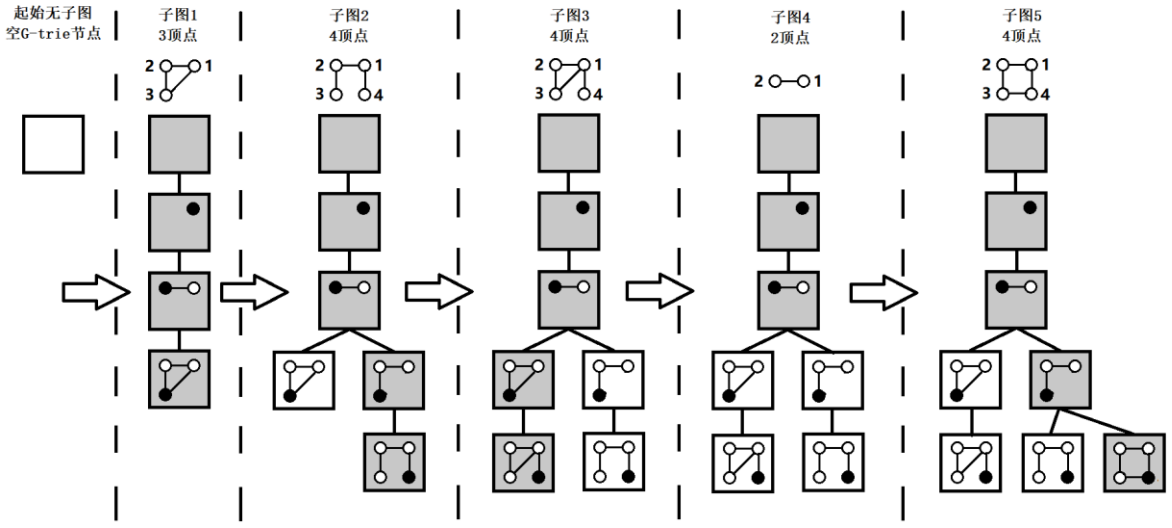


图 4-5 向 G-trie 中插入 5 个无向子图示例

2.2 算法流程

如上文所言，G-tries 算法主要分为两步：从待挖掘的子图集合构建对应的 G-trie

数据结构、使用 G-trie 查找对应的子图。我们将从这两个步骤所对应的详细流程入手介绍 G-tries 算法，并说明此算法优于其他算法的原因。

首先是构建 G-trie 的过程——表 4-1 程序 createGTrie(S_G)。我们遍历待查找的子图集合（往往是固定顶点数的所有可能子图集合），为每个子图执行一个插入 G-trie 的过程——表 4-1 程序 INSERT(*Node*, *Str*, *Cond*, *k*, *size*)。此时我们需要一个规范形式，以确保无论子图插入的顺序如何，最终构建的 G-trie 都具有唯一性，也就是说每个子图插入 G-trie 的路径都是固定的。规范形式根本上是为了解决图同构的问题，所以它的使用也保证了生成的 G-trie 拥有更多常见的子结构，可以节约存储空间、缩短搜索时间。在 G-tries 算法中，我们使用 GTCanon 来实现。它是一种定制规范形式，具有三大特点：连接性（G-Trie 中的路径将始终定义图形的联系）、压缩性（尽可能多地产生常见的子结构）、约束性（子图第一顶点尽可能多地连接，以便高度约束大的图中可能的匹配节点）。这种规范形式的使用把同构的子图归纳为同种模式，规定了子图插入 G-trie 时节点的顺序，保证了无论插入子图的顺序怎样变化，最终形成的 G-trie 始终唯一。

表 4-1 从子图集合创建 G-trie 的算法

算法 2-1 从一组子图集合中创建 G-Trie

输入： 子图集合 S_G
确保： 把 S_G 中的所有子图插入 G-Trie *T* 中

```
1: 程序 createGTrie(S_G)
2:     T := 空的 G-Trie
3:     for all 集合 S_G 中的图 G do
4:         Str := 规范形式的 G
5:         Cond := G 的对称破坏条件
6:         INSERT(T.根点, Str, Cond, 0, |V(G)|)
7:     过滤条件 T
8:     return T
9: 程序 INSERT(Node, Str, Cond, k, size)
10:    把 Cond 中的相关条件添加到 Node 中
11:    if k = size then
12:        将 Node 标记为图形的终端节点
13:    else
14:        for all Node 的子节点 c do
15:            if c.代表的子图 = k-顶点 Str then
16:                INSERT(c,Str,Cond,depth+1,size)
17:                return
18:        c := 新的 G-Trie 节点
19:        c.代表的子图 = k-顶点的 Str
20:        INSERT(c,Str,Cond,depth+1,size)
```

在介绍查找子图频次前，我们需要解决子图对称性的问题。同一个子图由于存在对称性（如三角形结构），在搜索时会重复多次出现。这占用了大量搜索时间，也造成了存储空间的浪费，所以我们需要使用对称性破坏条件来防止这种情况的发生。此对称破坏条件在构建 G-trie 时也有使用。如图 4-6 的示例，为了对三角形的子图进行计数并仅有效地查找一次，我们引入了{a<b, b<c}的对称破坏条件。我们在一个网络图中将每个顶点赋予一个标号，在这幅图中存在一个三角形的模式。在全图遍历三节点子图时一共有六种可能的方式遍历到这个三角形，而有效的遍历仅为第一次。为了筛掉其他五种重复遍历，我们把添加节点的标号顺序定义为{a, b, c}，当且仅当{a, b, c}满足对称破坏条件{a<b, b<c}时，我们才执行遍历操作，这样就可以大大节约遍历时间。

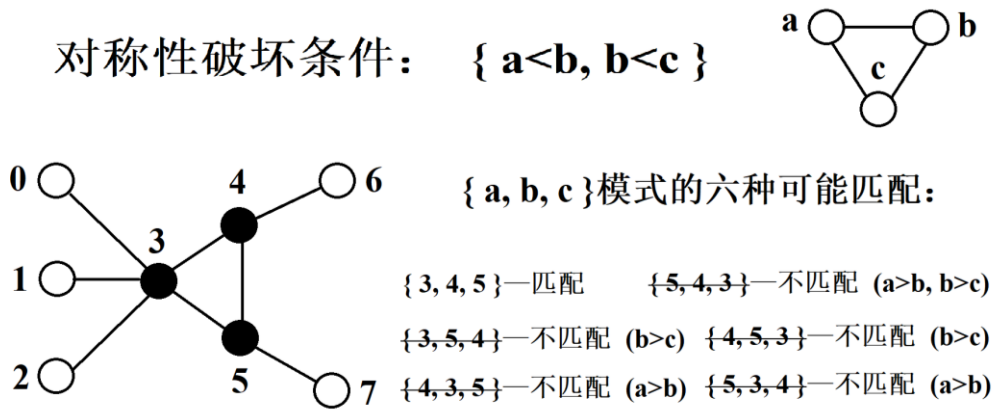


图 4-6 对称性破坏条件示例

最后是使用 G-trie 查找网络图中子图频次的过程——表 4-2 程序 queryGTrie(*T*, *G*)。该算法采用的搜索方法是普遍使用的递归回溯，在程序 queryNode(*Node*, V_{used}, *G*)中，找到当前子图匹配的部分，对应到 G-trie 中的某一个节点（从根节点递归）。然后由该节点开始，生成一组候选的节点。如果该节点是叶子节点并满足对称条件，那么我们就找到了该子图的一次出现。如果这个节点还有子节点，那么把它再作为新一轮递归的起始重复以上过程，直到整个网络图被搜索完成。

表 4-2 使用 G-trie 查找子图频次的算法

算法 2-2 查询网络图中子图的频次

输入： G-Trie *T* 和图 *G*
确保： *G* 的子图在 *T* 中出现过

```
1: 程序 queryGTrie(T,G)
2:     for all T.根节点的子节点 c do
3:         queryNode(c,∅,G)
4: 程序 queryNode(Node,V_used,G)
5:     V_cand := 匹配 Node 的 V(G)的候选
6:     for all v ∈ V_cand do
7:         if isTerminal(Node)∧
               conditionsOk(Node) then
8:             foundOccurrence(V_used∪v)
9:         for all Node 的子节点 c do
10:            queryNode(c,V_used∪v,G)
```

最影响搜索速度的步骤在候选节点生成的位置。为了提高运行效率，算法对此部分也进行了一定的优化：（1）V_{used} 从图表的较小邻域和与当前 G-Trie 顶点的连接中选取了较少的初始候选节点;（2）存储在节点上的对称破坏条件限制了可能的候选节点的间隔;（3）只有那些与 V_{used} 有联系的候选节点可以被保留。

2.3 使用 G-tries 挖掘带属性的网络

G-tries 算法同样可以用于挖掘带属性的网络[4]，包括网络顶点的属性和边的属性。如图 4-7 所示，对于点和边有多种属性的子图，在插入 G-trie 时只需要区分并构建不同的节点即可，在构建 G-trie 和查找子图频次的方法上都与无属性网络相同。同理对于有向图，我们也可以使用同样的方法挖掘网络主题。需要注意的是，随着网络

属性的增加，子图的个数也将呈现出指数增长。在属性值超过 10 个以后，无论在构建 G-trie 还是在搜索子图频次中，算法运行时间都会变得无法接受。而我们的告警数据属性个数都在几十以上，而对于单个簇的属性却相对比较单一。所以在实际应用中我们往往不会穷举所有属性的全部子图，而是把属性的统计量作为一个单独的特征输入分类器。这样我们可以在保留网络属性的条件下大大提高了搜索子图频次的时间。

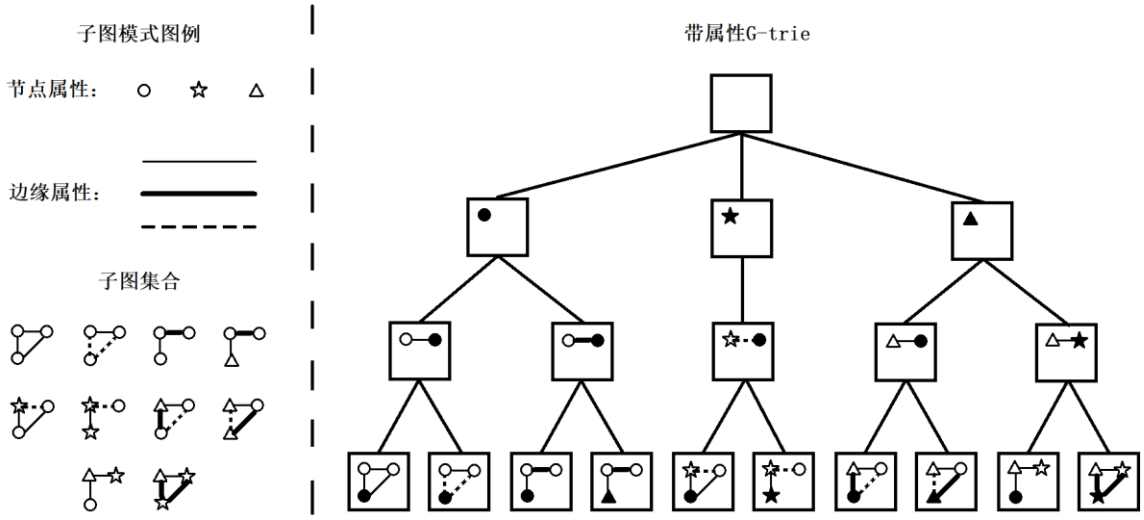


图 4-7 使用 G-tries 算法挖掘带属性网络示例

3. 告警数据关联融合

3.1 关联方法

由于网络的子图频次受到随机因素影响比较大，在实际操作中我们不能仅仅使用主题频次作为分类手段。为了排除随机干扰，我们往往会构造顶点数和顶点出入度与原网络相同的随机网络，并且将随机网络中子图出现的频次与原始网络对比。现在普遍采用的消除随机误差的变量是子图频次的 Z-分数[2]，即标准分数，其定义见式 1-4。主题 m 的 Z-分数值为原图中该主题出现的频次与随机网络中出现频次的平均值的差除以随机网络频次的标准差。

$$Z_m = \frac{f_m - \overline{f_m^{rand}}}{\sigma_m^{rand}} \qquad (4\text{-}7)$$

式中：Z_m——主题 m 的 Z-分数；f_m——原图中主题 m 出现的频次；$\overline{f_m^{rand}}$——随机网络中主题 m 出现频次的平均值；σ_m^{rand}——随机网络中主题 m 频次的标准差。

我们所采取的分类方法参考主流的使用 Z-分数消除随机误差的方式，结合应用场景，将 Z-分数大于某一阈值的主题提取出来，使用其出现频次的百分比作为分类

的依据。这种方法既消除了随机干扰，又将网络所独有的主题信息提取了出来，使得分类效果具有可解释性。在实际操作中考虑到计算成本，我们对网络的三顶点主题进行了挖掘。如图 1-3 所示，三顶点有向图总共有 13 种类型，将它们的出现频次归一化作为特征向量即可使用经典的分类方法进行处理。同时我们也考虑每个告警簇的个性特征，如告警厂商、告警事件、关键 IP 等等，结合带属性的子图挖掘算法，把分类效果又提升了一个档次。

我们所获得的原始告警数据量大且多源。如图 4-8 和 4-9 显示，告警数据平均每天在 3~5 万条左右，一天内最多高达 19 万条；告警数据来自 6 个不同的安全设备厂商，内外网各占约 50%。这就需要我们使用数据归并的方式将各种告警进行关联融合分析。在上节告警簇频繁子图挖掘中我们发现了几种典型的簇。在进行相似拓扑模式匹配时，我们发现有些模式在新旧两个簇中拥有相同的告警厂商和类型；而有些模式仅仅是拓扑结构相似，具体告警信息却有所不同。我们使用频繁子图 Z-分数指标来描述告警簇主题特征，同时引入告警类型、关键 IP 等信息，对现有数据进行了关联融合分析，找到了一些很有趣的告警模态。这些模态在拓扑结构上相似，同时具有相同的告警信息。在不断发现与总结典型模式的基础上，我们系统的实战能力会不断提高。

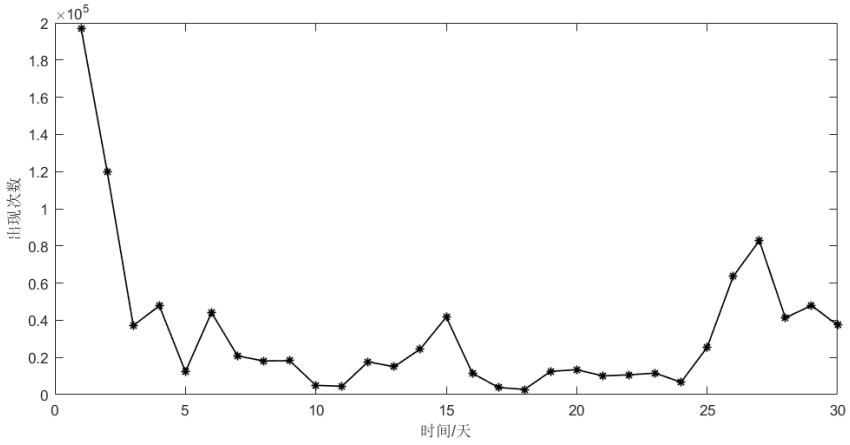
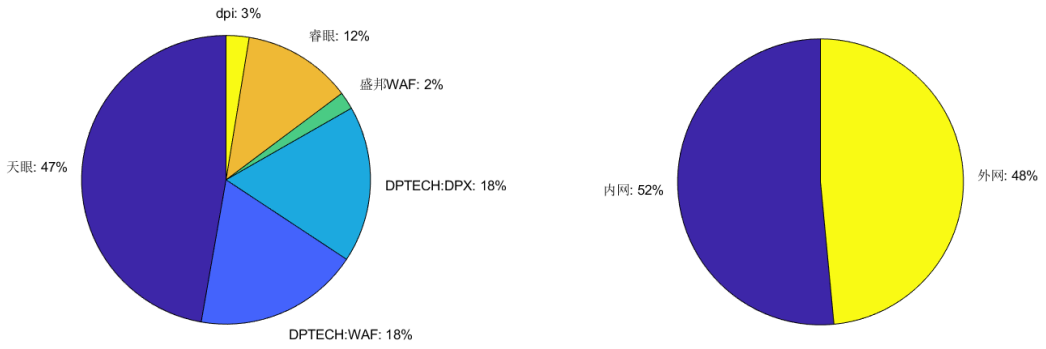


图 4-8　2021 年 4 月每天告警数量变化趋势图

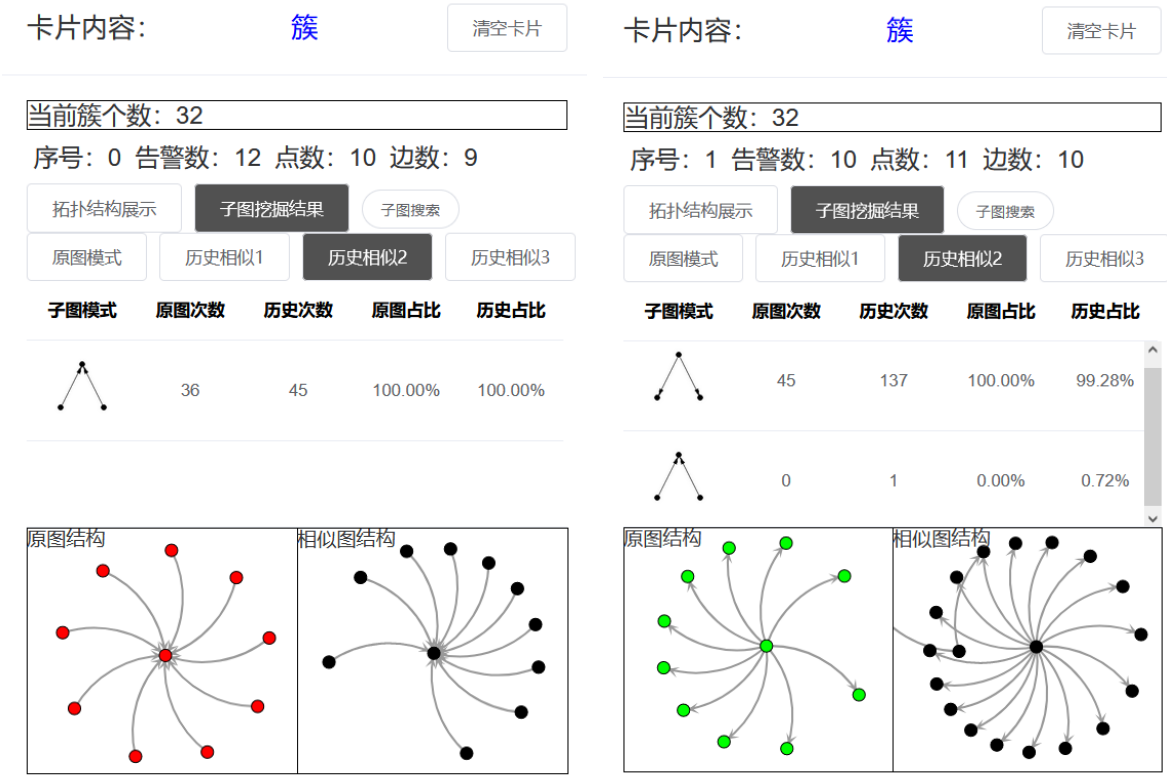


(a) 告警设备厂商占比　　(b) 内外网告警占比

图 4-9 全部告警数据占比分布

3.2 相似告警模态发现

在上文介绍可视化界面时我们已经提到了展示详细告警信息的卡片，不过它的功能远不止展示信息，在相似告警模态发现中此卡片也起到了重要作用。在可视化界面双击需要分析的簇的任意点即可展开簇分析卡片，可以对簇进行拓扑查看、子图模式挖掘和相似模态匹配等操作。如图 4-10 展示的两个典型簇模式的相似匹配，在卡片中选择子图挖掘即可使用 G-tries 算法进行频繁子图模式的计算，将返回的计算结果列为表格。同时结合 IP、告警类型等信息，在历史上搜索相似的告警簇。一旦找到相似匹配，就在卡片中展示当前簇和历史相似簇的拓扑对比。点击相关点和边可以查看二者的详细信息，便于我们快速确认告警簇的属性并进行处理。



(a) 汇聚型告警簇　　(b) 发散型告警簇

图 4-10 相似告警模态发现界面

4. 总结

4.1 高危报警的快速关联挖掘与定位

由于原始信息数量巨大，要实现高危报警的快速定位依赖于高效的后端算法以及高性能的前端框架。我们利用建立的黑名单数据库对高危模式、高危 IP 进行标记，一旦发生匹配，我们的系统可以在数秒内发现并产生报警。在告警数量极大、类型复杂的情况下也能对告警链进行及时准确的识别。我们利用现有数据，分别对安全演习和日常运行期间一天内每小时的告警数据进行了测试。如表 4-3 所示，安全演习期间平

均每小时的告警数量约为日常情况下的 5 倍，而我们系统的运行时间仅增加了 1 秒。总体来说无论何种业务场景，系统都可以在几秒内实现高危模态的快速识别与定位。

表 4-3 两种业务模态下系统前后端运行时间对比

业务模态	每小时告警数	平均计算时间	平均渲染时间	平均总时间
安全演习	5003	4.185 秒	0.4 秒	4.585 秒
日常运行	1058	3.51 秒	0.074 秒	3.583 秒

4.2 误报的批量可解释过滤

对于低风险模式与误报模式，我们把它们以事件簇的形式统一处理，在实现子图模式与关键信息匹配的基础上对告警簇进行分类。如图 4-11、4-12 显示，无论是安全演习期间还是在日常运行过程中，我们都将待处理的告警信息降低了 1~2 个数量级。

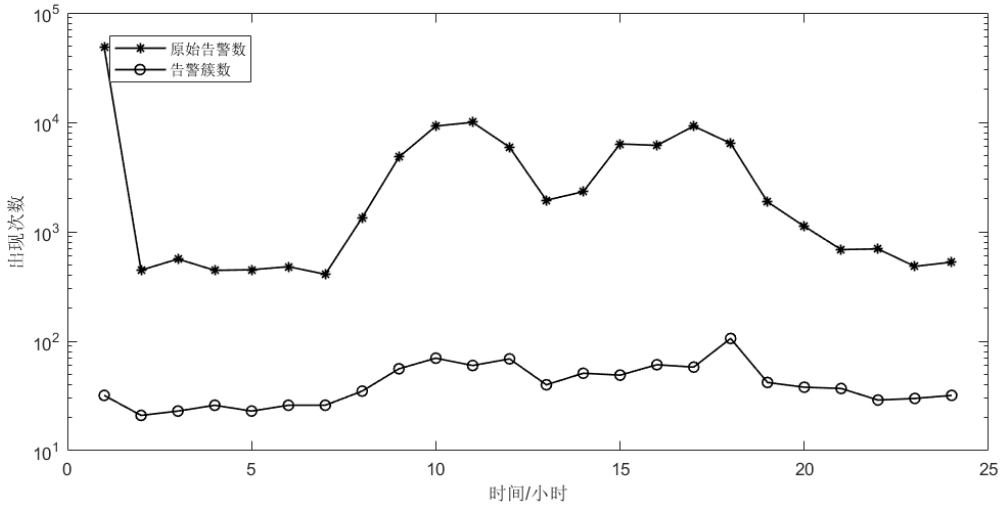


图 4-11 安全演习期间一天内原始告警与告警簇数量对比（对数纵坐标）

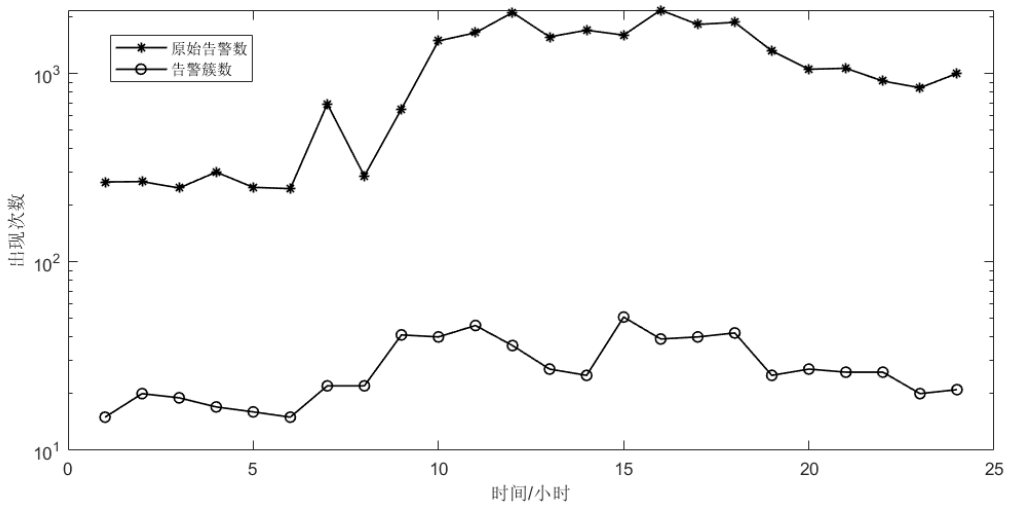


图 4-12 日常运行期间一天内原始告警与告警簇数量对比（对数纵坐标）

参考文献

[1] 杨安, 孙利民, 王小山, 等. 工业控制系统入侵检测技术综述[J]. 计算机研究与发展, 2016, 53(9):2039-2054.

[2] Fovino I N, Carcano A, De Lacheze Murel T, et al. Modbus/DNP3 state-based intrusion detection system[C]. 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, WA, Australia, April 20-23,2010.

[3] Yang Y, McLaughlin K, Littler T, et al. Rule-based intrusion detection system for SCADA networks[C]. 2nd IET Renewable Power Generation Conference (RPG2013), Beijing, China, Sept. 9-11, 2013.

[4] Goldenberg N, Wool A. Accurate modeling of Modbus/TCP for intrusion detection in SCADA systems[J]. International Journal of Critical Infrastructure Protection, 2013, 6(2):63-75.

[5] Mitchell R, Chen I. Behavior-rule based intrusion detection systems for safety critical smart grid applications[J]. IEEE Transactions on Smart Grid, 2013, 4(3):1254-1263.

[6] Fadlullah Z M, Fouda M M, Kato N, et al. An early warning system against malicious activities for smart grid communications[J]. IEEE Network, 2011,25(5):50-55.

[7] Hong J, Liu C, Govindarasu M. Integrated anomaly detection for cyber security of the substations[J]. IEEE Transactions on Smart Grid, 2014,5(4):1643-1653.

[8] Zhang Y, Wang L, Sun W, et al. Distributed intrusion detection system in a multi-layer network architecture of smart grids[J]. IEEE Transactions on Smart Grid, 2011, 2(4):796-808.

[9] Moghaddass R, Wang J. A hierarchical framework for smart grid anomaly detection using large-scale smart meter data[J]. IEEE Transactions on Smart Grid,2018, 9(6):5820-5830.

[10] Lipčák P, Macak M, Rossi B. Big data platform for smart grids power consumption anomaly detection[C]. 2019 Federated Conference on Computer Science and Information Systems (FedCSIS), Leipzig, Germany, Sept. 1-4, 2019.

[11] Marino D L, Wickramasinghe C S, Rieger C, et al. Data-driven stochastic anomaly detection on smart-grid communications using mixture Poisson distributions[C]. IECON

2019 - 45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, Portugal, Oct. 14-17, 2019.

[12] Girvan M , Newman M E . Community structure in social and biological networks[J]. Proc Natl Acad, U S A, 2002, 99(12):7821-7826.

[13] Gregory S . Finding overlapping communities in networks by label propagation[J]. New Journal of Physics, 2009, 12(10):2011-2024.

[14] Blondel V D , Guillaume J L , Lambiotte R , et al. Fast unfolding of communities in large networks[J]. Journal of Statistical Mechanics Theory & Experiment, 2008.

[15] Xie J , Szymanski B K , Liu X . SLPA: Uncovering Overlapping Communities in Social Networks via A Speaker-listener Interaction Dynamic Process[J]. IEEE, 2012.

[16] Gregory S . Finding overlapping communities in networks by label propagation[J]. New Journal of Physics, 2009, 12(10):2011-2024.

[17] Kok J N , Koronacki J , Ramon L , et al. An Algorithm to Find Overlapping Community Structure in Networks[C]// European Conference on Principles & Practice of Knowledge Discovery in Databases. Springer-Verlag, 2007:91-102.

[18] M Husa´K*†, MC Erma´K*†, M Lasˇ Tovicˇ Ka*†, et al. Exchanging security events: Which and how many alerts can we aggregate?[C]// 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). IEEE, 2017.

[19] Macedo M, Galo J, Almeida LD, et al. Demand side management using artificial neural networks in a smart grid environment[J]. Renewable & Sustainable Energy Reviews, 2015,41:128-133.

[20] Bhat Ta Charjee PS, Fujail A, Begum SA. A Comparison of Intrusion Detection by K-Means and Fuzzy C-Means Clustering Algorithm Over the NSL-KDD Dataset[C]// 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC).IEEE, 2017.

[21] Peng Ning, Yun Cui, Douglas S. Reeves. Constructing attack scenarios through correlation of intrusion alerts[P]. Computer and communications security, 2002.

[22] 冯学伟，王东霞，黄敏桓，李津．一种基于马尔可夫性质的因果知识挖掘方法［J］．计算机研究与发展，2014，51(11)：2493-2504．

[23] Zhou Y, Cui J, Liu Q. Research and Improvement of Intrusion Detection Based on

Isolated Forest and FP-Growth[C]// 2020 IEEE 8th International Conference on Computer Science and Network Technology (ICCSNT). IEEE, 2020.

[24]鲁显光，杜学绘，王文娟．基于改进 FP growth 的告警关联算法［J］．计算机科学，2019，46(08)：64-70．

[25]朱亮，陆静雅，左万利．基于用户搜索行为的 query-doc 关联挖掘[J]．自动化学报，2014 (08)：121-133．

[26]张仁斌，吴佩，陆阳，等．基于混合马尔科夫树模型的 ICS 异常检测算法[J]．自动化学报，2020，46(1)：127-141．

[27]梅御东，陈旭，孙毓忠，等．一种基于日志信息和 CNN-text 的软件系统异常检测方法[J]．计算机学报，2020，043(002)：366-380．

[28]Haas S , Wilkens F , Fischer M . Efficient Attack Correlation and Identification of Attack Scenarios based on Network-Motifs[C]. 2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC), 2019, pp. 1-11. doi: 10.1109/IPCCC47392.2019.8958734.