大连理工大学本科毕业设计（论文）

基于威胁情报基础库的关联关系可视化

Visualization of Association Relationships Based on Threat Intelligence Base

学 院（系）：软件学院

专　　业：软件工程

学 生 姓 名：蔡真真

学　　号：201392031

指 导 教 师：李豪杰

评 阅 教 师：王智慧

完 成 日 期：2017 年 6 月 6 日

大连理工大学

Dalian University of Technology

摘　要

威胁情报数据分析是网络安全防护及网络攻击追踪溯源的重要前提，但传统分析方式并不足以反映复杂网络数据的真实情况，只有充分结合专家分析模式与交互式可视化技术，才可以有效帮助分析人员更好的理解网络数据复杂的关联关系。现有的可视化成果存在交互性不够丰富有效、节点布局时间较长以及图的布局效果差等缺点。本文设计并实现了威胁情报多源异构数据的交互式关联钻取技术，利用可视化的方式呈现数据中隐含的信息与威胁的发展规律，提升安全分析人员对威胁情报的解读效率，进而辅助决策。论文的主要工作包括以下几个方面：

（1）总结归纳威胁情报可视化展示需求：调研国内外知名威胁情报厂商以及学术界在威胁情报可视化展示和分析技术中的研究成果，从多维度、多源异构数据节点布局以及交互策略方面优化本论文的威胁情报可视化方案。

（2）提出威胁情报数据关联关系可视化布局算法：针对威胁情报数据多维度、多源、数据异构的特性，采用单级力引导布局算法，引入模拟退火算法有效避免节点无用震荡，使用 D3.js 实现多源异构数据的交互式关联钻取和一件溯源功能。从提高算法效率和优化布局效果两个方面对算法进行改进。

（3）搭建威胁情报可视化系统：前端使用 vue.js 搭建数据驱动的 web 界面渐进式框架，vuex 做状态管理，vue-router 做路径切换，vue-resource 做数据通信，WebSocket 做全双工通信，D3.js 实现可视化展示与交互，element-UI 实现组件快速开发。

实验结果表明，本论文通过交互式关联钻取可视化分析技术实现了针对威胁情报数据的逐层关联钻取、基于力引导图的多源异构数据异步加载，并通过力引导-退火算法提高图形布局稳定的效率。

关键词：威胁情报；可视化；关联钻取；力引导-退火算法

Visualization of Association Relationships Based on Threat Intelligence Base

Abstract

The analysis of threat intelligence is an important prerequisite for network security protection and network attack tracing. However, the traditional analysis method is not enough to reflect the real situation of complex network data. Only by combining expert analysis mode and interactive visualization technology can help the analysis better understanding of the complex relationship between network data. The existing visualization results are not enough rich and effective, the node layout time is longer and the layout of the figure is poor. In this paper, we design and implement the interactive associated drilling technology of threat intelligence multi-source heterogeneous data, and use visualization to show the development of implicit information and threat in the data, to improve the efficiency of the security analyst by interpreting the threat intelligence. The main work of the paper includes the following aspects:

（1）Summarize the visualization requirements of threat intelligence: investigate the research results of well-known threat intelligence vendors and academics in threat intelligence visualization and analysis technology, and optimize the thesis from multi-dimensional, multi-source heterogeneous data node layout and interactive strategy.

（2）Proposed threat data association relationship visualization algorithm: Aiming at the characteristics of multi-dimensional, multi-source heterogeneous data of threat intelligence, a single-level force-directed layout algorithm is introduced. the introduction of simulated annealing algorithm effectively avoid the node useless shock, and the use of D3.js achieve heterogeneous data interactive drill and a traceability function. The algorithm is improved in two aspects: optimizing the layout effect and improving the efficiency of the algorithm.

（3）Build a threat intelligence visualization system: front-end use vue.js to build data-driven framework, vuex to do state management, vue-router to do path switching, vue-resource to do data communication, WebSocket to do full duplex communication, D3.js to achieve visual display and interaction, element-UI to achieve rapid development of components.

The experimental results show that this paper uses the interactive correlation drilling visualization analysis technology to realize the hierarchical correlation drilling for threat intelligence data, load asynchronously multi-source heterogeneous based on force-directed graph, and improve the stability of graph layout by Force-SA Algorithm.

Key Words：Threat Intelligence；Visualization；Associated Drilling；Force-SA Algorithm

目　　录

1 绪论

1.1 课题背景

在信息全球化的今天，继陆、海、空、天之后网络空间逐渐发展为第五大战略空间，是影响国家安定、社会安全、经济腾飞的基础和关键。随着网络空间大规模扩张，我国关键基础设施以及政府相关部门网站受到攻击的事件频频发生，各种网络攻击和网络威胁呈现出一种持续性和放射性的发展趋势且日益严峻。传统的以特征检测为手段的静态防御方式，例如防火墙、防病毒软件、入侵检测等，已经难以对抗 APT 攻击、0day 漏洞等持续变化的攻击手段。对此，“威胁情报”应运而生。2013 年，Gartner[1]提出了威胁情报自适应安全体系，被业内公认为信息安全行业未来发展趋势，许多企业乃至国家已将威胁情报体系的建设纳入其安全战略规划之中。

在威胁情报共享方面，各自为政地自主开展情报收集工作不但限制了情报在各组织之间的积极流通，而且大大增加情报采集成本，这极易导致形成信息孤岛。众所周知，威胁情报的发展是一个全面且长期的过程，也是一个各组织间积极协同配合的过程。只有将横向时间与纵向空间内的情报充分融合，才能形成完整的威胁情报系统；只有将对威胁情报分析的重视提升到与一定高度，才能充分发挥情报数据的真实价值，真正体现威胁情报的重要决策辅助作用和商业价值[2]。

在威胁情报可视化方面，网络可视化技术巧妙地运用了人类视觉感知系统，将网络结构数据以图形化的方式展示出来，给用户直观而丰富的视觉效果，从而帮助用户更好的理解数据。这不仅能够帮助人们更直观地了解网络结构，而且也能帮助人们深度挖掘网络内部隐藏的重要的信息。比如，使用网络安全可视化设计能够在短时间内形象而直观的使安全分析人员感知到攻击源、攻击目标以及受攻击的重灾区等信息，从中将事件进行分类，并快速识别风险，甚至能够对攻击未来发展趋势进行预测。所以通过可视化分析技术对典型的威胁情报数据进行可视化设计，能够将威胁情报信息更直观、形象地展示出来，减少相关安全人员分析与思考的时间，使其更快地做出更明智的决策。

但是目前对于威胁情报数据的可视化设计还未形成十分有效的通用性方法，网络安全领域使用可视化技术进行数据分析，但传统分析方式并不足以反映复杂网络数据的真实情况，只有提供与分析技术相结合的交互式可视化界面，才可以帮助分析人员更好的理解网络数据复杂的依赖关系，减少分析与思考的时间，增强网络安全防御能力，为网络安全提供有力的信息保障，尤其是对于急需高级持续性威胁攻击（APT）检测或者定向攻击检测方案的今天。

1.2 研究意义

目前，我国威胁情报建设体系缺乏上层组织与顶层架构，即尚未形成“国家队”。虽然各个部门都在自主筹建安全检测防御平台，但缺乏一个由安全厂商、数据提供商、研究机构、用户等多方共同参与的合作体系。

本文所依托的威胁情报可视化系统，隶属于中国科学院战略性先导科技专项“面向感知中国的新一代信息技术研究”-基于威胁情报的溯源分析研究任务中的部分研究成果。是国家某重大工程项目的原型系统，致力于联合相关单位建立数据采集汇集和共享开发利用机制，构建国家网络空间威胁情报大数据共享开放平台。

针对威胁情报做可视化展示与分析的优点在于能够让分析人员在威胁情报数据中发现未知线索、规律以及异常情况[3]。例如当用户试图分析一个安全事件时，往往需要追踪攻击的源头，找到实施攻击的黑客及其组织，或者通过攻击的 IP 地址找到实施攻击行为的黑客的真实信息，威胁情报的可视化能够给用户提供和这个安全事件相关联的情报信息，通过一层一层地向下钻取，追踪溯源到攻击者的真实社会信息。可见，将威胁情报数据进行可视化展示能够帮助安全分析人员高效地处理已经发生或正在发生的威胁。

现有的可视化成果存在交互性不够丰富有效、节点布局时间较长以及图的布局效果差等缺点。本文主要围绕构建威胁情报数据的通用性可视化展示方法，针对威胁情报数据的多维度、多源、数据异构的特性来开展可视化展示的研究工作。主要的工作重心在于威胁情报多源异构数据关联关系的可视化设计及实现。通过直观的可视化图形和“傻瓜式”的交互，使得用户能够直观、全面地了解和分析安全事件存在的关联关系，最终达到追踪溯源的目的。

1.3 相关工作

随着网络技术的飞速发展，信息安全领域所面临的威胁形势日益严峻。传统的被动式防御技术已不能满足对高级持续性威胁攻击、0day 漏洞等新型网络威胁的防护，针对于此，“威胁情报”应运而生。如今，学术界和工业界都专注于提出检测和防御各种高级攻击的网络安全可视化解决方案。

在学术界，D.M.Best[4]等人提出了网络防御可视化方面的七个关键挑战，作者认为，尽最大的努力去开发网络可视化技术并非是为了吸引关注。据作者观点，独特的解决方案和美观的图形可视化开发的主要目的是为了影响用户。只有清楚地了解用户的需求和解决他们的需求，才是成功开发可视化平台的关键因素。D.M.Best 提到的七个关键挑战也是本论文在总结归纳威胁情报可视化展示需求的重要参考依据，如何帮助安全分析人

员提升对威胁情报的解读效率和辅助决策一直是本文可视化设计方案的关键。D.Lee[5]等人提出了使用可视化技术加速恶意代码模式分析过程。他们的工作中心在于恶意软件分析，并为可视化提供了良好的应用情境，这是识别和提取未知的恶意软件模式所需要的。S.Chen[6]提出了一个在线协作和探索性分析工具，将其命名为 OCEANS，以此帮助网络管理员和安全分析人员分析网络流量和日志数据。OCEANS 提供多层次的可视化，也提供有关 IP 连接的时间概况，并允许参与者协同配合寻找事件源头和定位攻击。另一个可视化分析工作是 Fischer[7]等人提出 Nstreamaware，它是一个分析数据流的系统，允许分析器与系统进行交互来控制聚类过程，以此将数据流的大小减小到有意义的分段。该系统能够组合不同类型的数据，实现大数据分析和态势感知功能，以此增强网络安全防御能力。当然，也有不少科研机构发布了威胁情报共享与分析的原型系统，例如，J.Vijaya Chandra[8]等人提出了在云平台上根据社会工程学部署的情报防御系统，该系统可以防卫高级持续性威胁攻击。Carvalhoto[9]提出了针对网络威胁的实时检测和可视化平台，该平台为遍布全球超过 270 个被感染的独立 IP 提供可视化展示和告警分析。

在工业界，国外已初步形成相对完整的威胁情报分工体系和产业链。Symantec、FireEye、IBM、CrowdStrike、Webroot、ThreatBook、iSight Partners 等专业组织和厂商已经推出了相应的解决方案和服务，国内一些技术纯熟的厂商和组织机构也积极投入到威胁情报相关产品的研发和生态系统建设过程中。以下为国内外部分知名厂商在威胁情报可视化展示上的相关成果：

图 1.1　Norse 公司的网络实时攻击展示地图

（1）Norse Attack Map：是由 Norse 公司提供的全球黑客战实时攻击情况可视化展示网页，被称作“上帝的视角”，该公司在全球各地放置了超过 800 万个攻击传感器，将这些传感器伪装成最容易受到网络攻击的设备，例如：PC、Mac 电脑，银行 ATM 机等。当传感器受到攻击的时候，攻击数据就通过 VPN 通道传回公司后台数据库。当用户访问网页时，向后台发送请求，后台实时推送最新的攻击数据，并映射在地图上。该界面充分展现了与各个攻击相关的多角度、多维度信息，安全分析人员能够直观地通过地图的实时攻击效果了解当前世界网络攻击实况。

（2）Kaspersky Cyberthreat Real-time Map：卡斯巴基实验室于 2014 年发布了这款展示全球实时网络攻击情况的互动地图，如图 1.2 所示，每个点表示网络攻击中的攻击者和受害者，每条线表示攻击路径，不同颜色代表不同种类的网络攻击，与 Norse Attack Map 不同之处在于它采用三维展示效果，具有更好的用户交互性，能够根据用户的不同需求呈现多样化的数据，更加迅速地定位攻击来源。

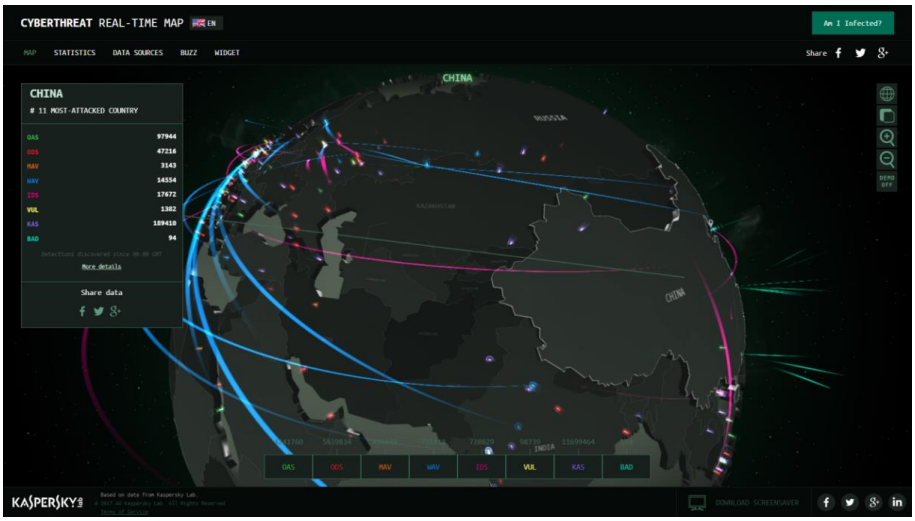

图 1.2　卡斯巴基实验室发布的实时网络攻击互动地图

（3）Threatcrowd：一款能让用户搜索与 IP、网站或者组织机构相关的威胁信息的情报搜索引擎。Threatcrowd 会从 Virustotal、malwr.com 等站点上采集威胁情报数据，同时它也能够支撑 MALTEGO 的转换，方便情报的分析和数据的关联。如图 1.3 展示的是对 IP>188.40.75.132 的多层关联钻取信息。在可视化展示之前，Threatcrowd 对关联数据进行过滤，将过滤后的信息以静态布局的方式展示出来，并不支持节点关联拓展和自动更新，这容易导致用户分析数据受限，不能发现一些潜在的的有价值的威胁信息。

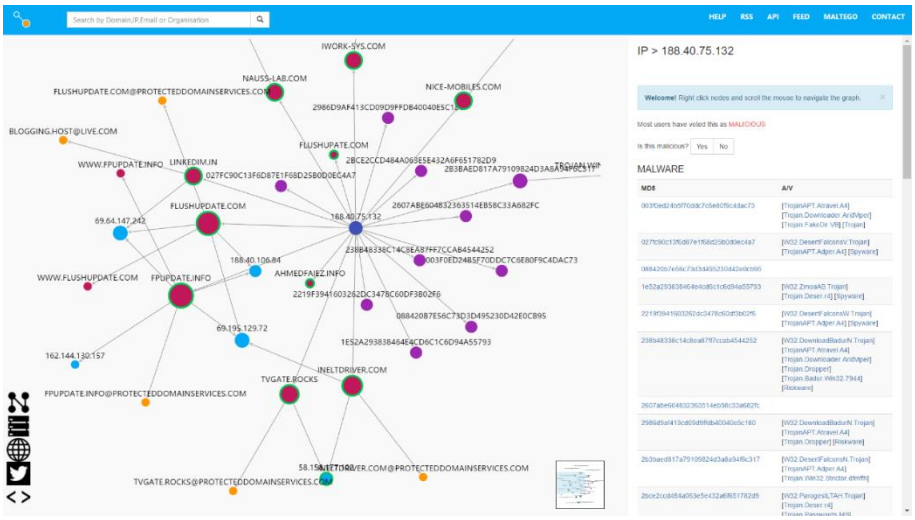

图 1.3　Threatcorwd 提供的 IP>188.40.75.132 的多层关联钻取信息

（4）VirusBook：是国内首家威胁情报公司——微步在线（ThreatBook）在 2016 年正式发布的国内第一款综合性的威胁分析平台，该平台致力于为全球的安全分析人员提供了一个便利的一站式威胁情报分析平台，用于进行包括事件检测、事件确认、影响分析、关联钻取及溯源分析在内的事件响应过程工作。如下图 1.4 所示，是对域名>manage-163-account.com 的两层关联钻取信息，可以看到与该域名相关的 hash 样本信息。

图 1.4　ThreatBook 对域名>manage-163-account.com 提供的可视分析

（5）Alice 威胁情报溯源平台：该平台的开发者为天际友盟，专注于威胁情报的聚合、分析、交换，致力于推动和建设全球化的威胁情报生态系统。如图 1.5 所示，该平

台结合自身专业的情报分析能力，依托于来自合作伙伴广泛且全面的威胁情报共享信息，为企业用户提供可靠的威胁数据查询以及追踪溯源服务。系统后台借助机器学习引擎对多源大数据进行处理，将威胁信息利用三度分析模型进行实时关联。

图 1.5　天际友盟威胁情报溯源平台提供的安全威胁情报态势图

总结上述威胁情报厂商可视化产品的特点：NORSE 和 Kaspersky 公司都是采用 2D 或者 3D 的地理位置信息实时反映世界黑客网络攻击实况，呈现给用户直观的效果以及真实的数据信息。ThreatBook、threatCrowd、天际友盟三家都使用 Cytoscape.js 这个开源图形库实现威胁情报数据节点关联关系的可视化，但普遍存在交互性不够丰富有效、节点布局效果差、节点无法逐层关联拓展的弊端。

1.4　主要研究内容以及贡献

本论文依托的项目是威胁情报可视化系统，在系统的实现过程中针对威胁情报多源异构数据的关联关系可视化展开研究工作。核心工作包含以下三个方面：

（1）总结归纳威胁情报可视化展示需求：调研国内外知名威胁情报厂商以及学术界在威胁情报可视化展示和分析技术中的研究成果，从多维度、多源异构数据节点布局以及交互策略方面优化本论文的威胁情报可视化方案。

（2）提出威胁情报数据关联关系可视化布局算法：针对威胁情报数据多维度、多源、数据异构的特性，采用单级力引导布局算法，引入模拟退火算法有效避免节点无用震荡，使用 D3.js 实现多源异构数据的交互式关联钻取和一件溯源功能。从提高算法效率和优化布局效果两个方面对算法进行改进。

（3）搭建威胁情报可视化系统：前端使用 vue.js 搭建数据驱动的 web 界面渐进式框架，vuex 做状态管理，vue-router 做路径切换，vue-resource 做数据通信，WebSocket 做全双工通信，D3.js 实现可视化展示与交互，element-UI 实现组件快速开发。

对威胁情报数据设计通用性的可视化方法，实现数据的逐层关联钻取和一件溯源功能，使得安全分析人员能够通过可视化展示方案，准确而高效地识别重要威胁信息，预测未来威胁发展趋势进而积极地做出预防措施。

1.5 文章结构安排

本论文对威胁情报数据提出了可视化的通用性设计方法及其系统实现方案，文章结构安排如下：

第一章是绪论，首先阐述了课题背景和研究意义，然后分析了国内外工业界和学术界在威胁情报可视化方面的研究成果，并对本文的研究内容及贡献进行了概括。

第二章介绍了威胁情报以及可视化技术，首先阐述了威胁情报的概念，并介绍了我国现如今在威胁情报建设方面与国外的差距和面临的挑战。然后，简单介绍图可视化的概念，以及图可视化面临的问题。最后，对单级力引导布局算法的发展历程和三大主流算法思想进行概述。

第三章详细阐述了关联关系可视化布局算法的实现，是本论文的重点。首先介绍威胁情报的可视化需求和可视化框架的设计以及可视化流程。然后详细阐述了关系图布局算法的实现与优化过程，采用单级力引导布局算法进行布局，引入模拟退火算法有效避免节点震荡，从提高算法效率和优化布局效果两个方面对算法进行改进。

第四章是对威胁情报可视化系统进行设计，首先对可视化系统的功能需求进行分析，然后阐述可视化系统框架的设计原则和整个平台架构设计，最后对该可视化系统使用的前端技术进行详细介绍。

第五章是对可视化系统功能模块实现与测试，对用户登录、情报查询、可视分析、情报维护、情报统计五个功能模块的主体功能和实现机制进行详细介绍。本章的重心在于关联钻取、一键溯源功能实现以及对第三章提出的力引导-退火算法进行布局时间测试。

第六章总结了本文的贡献和不足以及提出了下一步需要完成的工作。

2 威胁情报与可视化技术

2.1 威胁情报概念及研究现状

2.1.1 威胁情报概念

目前，对于网络安全威胁情报的认知存在多样化，但普遍认为，威胁情报是指所有有助于用户避开网络攻击的有价值的信息。这些信息揭示和反映了网络中可能面临的威胁或者危险，以此帮助用户及时提出有针对性的应对措施。

国外知名咨询公司 Gartner 于 2013 年给出了威胁情报的相关定义：“威胁情报是一种基于证据的知识，它就网络资产可能存在或出现的风险、威胁，给出了相关联的场景、机制、指标、内涵及可行的建议等，可为主体响应相关威胁或风险提供决策信息。[10]”由此可见，威胁情报的目的是还原网络空间已经发生的入侵行为，形成线索来预测尚未发生的攻击。可以说，威胁情报的出现，促进了网络防御思想从基于漏洞为中心向基于威胁为中心的转变。

威胁情报的出现和发展被视为解决未来网络安全问题的关键。对威胁情报进行感知、共享和可视分析，主要目的在于采用多样化的技术来采集大规模碎片式的异常数据，对这些数据进行深度挖掘，从而形成有价值的威胁情报线索集合，然后使用事件关联、网络可视化、机器学习等技术手段分析已发生的攻击行为，在此基础上预测未来网络威胁态势，以此帮助用户制定高效的安全决策方案，提升网络安全防御能力。

2.1.2 威胁情报研究现状

自 2014 年开始，威胁情报已成为网络安全领域的热点话题。目前，国外已形成相对完整的威胁情报分工体系和产业链。在美国，威胁情报分析技术已被广泛认可，爱因斯坦计划项目的实施，网络天气地图的构建，网络威胁与情报整合中心的建立，以及《网络安全情报分享法案》等一系列法规的出台，都彰显了美国从国家层面和战略高度上对威胁情报分析的重视。

目前，我国的威胁情报体系发展相对落后，但机遇与挑战并存。在学术方面，威胁情报各阶段所涉及的学术成果散乱，缺乏有效的威胁情报的采集、分析、共享规范，其中，杨泽明[11]等人提出了通过攻击溯源的威胁情报共享利用研究；王慧强[12]等人提出了网络安全态势感知关键实现技术研究;王卓君[13]提出了一种用于威胁情报评估的数据分类算法研究；吕宗平[14]等人提出了基于攻击链和网络流量监测的威胁情报分析研究；管磊[15]等人提出了基于大数据的网络安全态势感知技术研究；李骏韬和施勇[16]等人提出了基于

DNS 流量和威胁情报的 APT 检测。上述对威胁情报的研究主要集中在情报共享和情报评估分析方面，威胁情报的可视化研究相对较少。

在工业界，尚未形成集数据采集、分析、转化、共享为一体的大型现代化威胁情报中心，缺乏一个面向政府、企业、设备厂商、个人以及民间组织提供威胁情报的综合服务平台。但是，由奇虎 360 牵头建设的威胁情报信息管理中心已在部分地区试运行；阿里云盾也在现有产品的基础上增加了威胁请把态势感知功能，涵盖了最新的威胁情报等一系列告警信息；网康慧眼云融入了威胁情报系统，通过情报地图和情报检测可以实时查看最新的网络攻击情况；天际友盟等 8 家公司正式成立了“烽火台”安全威胁情报联盟，推动了国内威胁情报生态圈的发展。

在学科建设方面，网络空间安全专业已经成为国家一级重点学科，各大高校和相关研究部门都积极投入到学科建设中，致力于为社会源源不断地输出网络安全人才。

2.2 图可视化概念及面临问题

2.2.1 图可视化概念

“计算机的目的不在于数据，而在于洞察事物。”著名的数学家海明曾这样说过。为了使计算机收集到的海量数据便于理解，我们需要一种高效快捷地探索知识的方法，即可视化技术。可视化技术起源于科学计算可视化，是 1987 年 2 月在美国科学基金会召开的研讨会上提出的新名词，在二十世纪 80 年代后期发展成为发达国家的一个新兴研究领域。

1736 年，自哥尼斯堡七桥问题解决之后，图的数学理论的相关研究越来越多，图的可视化理论随之发展。经过几百年的积累，在关系图的可视化研究领域，已经有了坚固的理论基础和一系列高效处理图数据的独立算法。可以说只要是有内部关系的数据都可以用图模型来表示，用点来表示实体，用边来表示实体之间的关系。

图可视化技术是信息可视化技术的一个分支，其目标是以形象的图来展示抽象的图数据，以帮助人们挖掘图数据中的有用信息[17]，该技术已被广泛应用，并衍生出许多可视化分析软件，例如 Tableau、Raw、ChartBlocks 等。虽然这些软件满足了用户部分可视化需求，但它们图形展示种类单一，可拓展性差，不支持不规则数据，并不能满足用户频繁变动的可视化需求。

2.2.2 图可视化面临的问题

随着社会科技的高速发展，图可视化技术愈加成熟，但仍然面临着以下几个方面的问题：

（1）复杂度问题：图可视化技术的算法复杂度极高，相当于一个 NP-hard 或 NP 完全问题，即使以牺牲空间或者其他代价来换取时间的策略改进算法，该算法的复杂度也达到了$o(kn^2)$以上。因此，图可视化技术并不适用于数据量规模过大的数据展示。

（2）可读性问题：图的可视化有多种布局算法，不同的布局算法具有不同的特点，因此，如果选择不合适的图可视化布局算法，就会降低分析人员对可视化图形的理解能力，影响工作效率，与可视化展示的初衷背道而驰。下图 2.1 通过 echarts3 编码展示了三种不同的图可视化布局算法对相同数据进行布局的可视化结果。可知，①图采用力引导布局算法，节点相互制约以达到力平衡，因此，整个布局图形显得比较均匀；②图采用圆形布局算法，具有对称性美感；③图采用随机布局算法，每个节点的位置都是无根据进行随机分布的，因此，整个布局图形看起来杂乱无章。

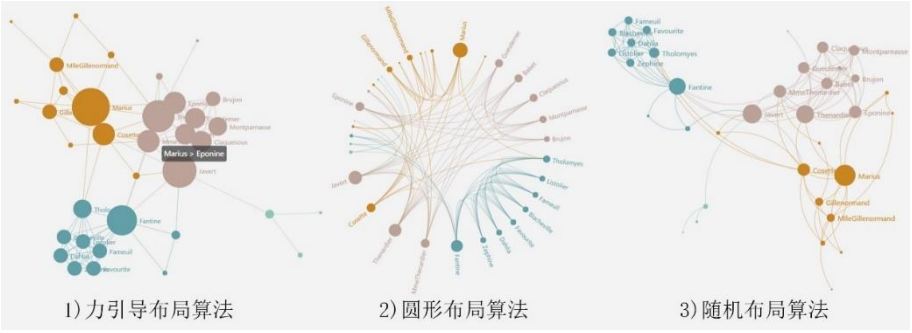

图 2.1　不同的图布局算法对相同数据进行布局的效果

（3）交互问题：用户交互是图形可视化技术的两大核心问题之一。在分析人员进行可视分析时，往往需要与系统进行交互，通过各种操作触发不同的监听事件，从而获取相应的反馈。然而，如何设计合理且简单的用户交互操作，是系统能否被广泛用户认可的关键。

（4）空间利用问题：不同的可视化布局算法不仅时间复杂度不同，同时对空间的利用率也迥异。随着数据量的增加，节点的布局更为密集，如何合理利用有限的空间，提高空间利用率，也是图可视化技术研究的方向之一。

（5）美观性问题：并非将数据以简单的图形方式进行展示就称作图形可视化，真正的图形可视化技术所产生的的布局图应该符合美学标准。

然而，在实际的图形布局中，并不能完全解决上述所有问题，例如，想要实现布局图形的美观性，就得提高图形算法的迭代次数，进而大大增加算法的时间复杂度，问题之间是相互矛盾的。因此，在实际应用中，我们只需考虑解决上述若干个问题中的一两个问题。例如，从图形的算法复杂度和布局效果的美观性两个方面进行考虑。

2.3 图布局算法综述

2.3.1 图布局美学标准

有上一小节可知，人们往往从图形的算法复杂度和布局效果的美观性两个方面来评价一个布局算法的优劣。其中，算法复杂度包含空间复杂度和时间复杂度，对于空间复杂度和时间复杂度可以通过逻辑运算量化，例如，用$O(kn^2)$表示。对于布局效果的美观性，目前，还没有一个统一的衡量标准。每个人的审美是不同的，评判事物的标准也迥异，因此，对图形美观性的判定并没有具体的指标。但是，业内普遍认为图可视化布局算法应该满足以下几个美学标准：

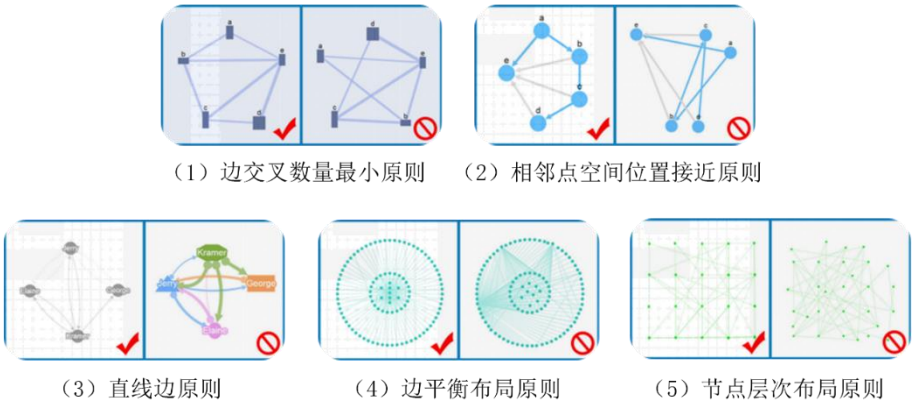

图 2.2 图布局算法五大美学标准

（1）边交叉数量最小原则。减少边的交叉数量可以更加清晰地展现图结构。如图 2.2①所示，节点相同位置进行布局，无边交叉的图结构比四边交叉的图结构更为清晰明朗。

（2）相邻点空间位置接近原则。如图 2.2②所示，将相邻接的节点尽可能地布局在相近的位置上，可以减少边的长度以达到更高的空间利用率。

（3）直线边原则。图的边使用直线比曲线更能清晰地展现图结构。

（4）边平衡布局原则。连接相同节点的多条边应尽量以该节点为中心进行布局。

（5）节点层次布局原则。节点应尽量布局在水平或垂直的不同层上。如图 2.2⑤所示，相同的节点进行布局，左图的层次感更强，图结构也更加明朗。

此外在进行图布局时，还要从可溯性和规律性两点进行考虑。

（1）可溯性是指节点之间的关联关系应该用边相连，形成路径。

（2）规律性是指在进行图布局时所选用的参考原则应始终如一。

在实际应用中的图布局算法要同时满足以上所有指标需要消耗大量的时间和空间，Helen Purchase[18]在评价关系图布局效果时，通过实验测试，对偏向于某一个指标布局的关系图的理解时间、理解错误率展开调查，进而对这五种指标的重要性进行了排序，发现减少交叉边数是最重要的评价指标；减少弯曲边数，能够最大程度体现对称性，但对于图的可读性有很小的作用；减小边弯曲度和最大程度正交化边对于理解图几乎没有任何帮助。

2.3.2 单级力引导布局算法

1984 年，Eades[19]提出了弹簧模型(Spring-Embedded Model)布局算法。该算法的基本思想是将图看成一个物理系统，借鉴物理学中的“弹力”，将关系图的边看成弹簧。布局开始时，给系统赋予某个初始运动状态，关系图的节点通过弹簧弹力的拉伸作用进行移动，直到整个系统的总能量减少到 0 或者接近 0 的某个数时，系统达到静止状态。在 Eades 的提出的弹簧模型布局算法中，并没有遵循物理学中的胡克定律，而是采用了自定义的弹簧受力公式。Edges 的弹力计算公式为：

$$F(v) = \sum_{(u,v)\notin E} F_r^u(v) + \sum_{(u,v)\in E} F_a^{(u,v)}(v) \qquad (2.1)$$

其中，Fr——斥力，与任意两点之间的欧式距离的平方成反比；

Fa——引力，与边的对数值和边的长度成正比。

该算法的时间复杂度为$o(kn^2)$，将终止条件设为最大循环迭代次数，当达到循环迭代次数时，算法终止。此外，Eades 为了降低算法复杂度，提出如下假设：节点的引力作用只存在于两个相邻节点间，斥力作用存在于所有非相邻节点对之间。

1989 年，Kamada[20]等人在弹簧模型布局算法的基础上进行改进，加入胡克定律，提出一种新的图布局算法——能量模型（Energy Model）布局算法。该算法通过降低系统总能量达到最小值来确定关系图每个节点的位置。该模型的能量公式如下：

$$E = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \frac{1}{2} k_{ij}(|\ p_i - p_j\ | - l_{ij})^2 \qquad (2.2)$$

其中，k_{ij}表示弹性系数，$|p_i - p_j|$表示节点$<i,j>$之间的距离，l_{ij}表示理想距离，是指两个节点之间路径的总数与关系图半径的比值。整个能量公式可以看成是求关系图任意两个节点$<i,j>$之间理想距离与实际距离之差的平方和，然后乘以弹性系数，当值达到最小时，系统的能量值 E 为最小值。

然而，要是能量值 E 达到全局最小化，是十分困难的，因此，Kamada 等人使用牛顿-拉普森算法对每个节点进行优化，通过公式来判断布局是否达到平衡状态，stress(G)的计算公式为：

$$stress(G) = \sum_{u<v} (||P_u - P_v|| - d_{uv})^2 / d_{uv}^2 \quad (2.3)$$

其中，d_{uv}——节点 u 与 v 最短路径，作为平衡状态下距离；

P_u——节点 u 的坐标；

P_v——节点 v 的坐标。

在进行每一次循环迭代时，都查找出使stress(G)减少最大的节点 w，沿着能量值变化减小的方向移动节点 W，循环直到算法结束。该算法实现起来比较简单，最大的特点便是逐个调整节点的位置，这样可以有效减少边交叉，但有可能只调整几个节点，就达到最大循环迭代次数。

此后，Fruchterman 和 Reingold[21]两人借鉴天体运动的思想，提出了力引导模型（Force-Directed Model）布局算法，又称作 FR 算法。根据物理学原理，原子内部粒子或天体彼此间会产生引力和斥力，在合力的作用原子或天体的运动，直到达到平衡状态或者能量耗尽，Fruchterman 和 Reingold 正是将关系图的节点看成是原子或者天体。两人借鉴 Eades 的弹簧模型布局算法，在它的基础上进行优化和改进，但与弹力模型不同的是，FR 算法中的相邻节点会产生引力的作用，同时所有的节点对之间还会产生斥力作用，每个节点在合力作用下运动，最终系统达到一个平衡状态。为了避免节点震荡以及提高布局效率，FR 算法引入了模拟退火的过程，但是并没有给出具体的实现公式。该算法在每一次循环迭代过程中都会经历了以下 3 个步骤：

（1）计算所有节点来自相邻节点的引力；

$$F_a = ||x_u - x_v||^2/k \quad (2.4)$$

（2）计算每个节点对之间斥力的大小；

$$F_r = \frac{k^2}{||x_u - x_v||} \quad (2.5)$$

（3）每个节点受到引力和斥力的合力作用而运动，运动的幅度受到某个值的限制，该值称为“温度”，随着迭代次数的增加，“温度”减小到 0 或者某个接近 0 的值，节点最终达到静止状态，系统能量达到平衡，图布局完成。

弹簧模型布局算法、能量模型布局算法以及力引导模型布局算法合称为单级力引导布局算法的三大基石。在之后的十几年里，国内外学者对这三种算法进行了大量的改进，主要从优化布局效果以及提高算法效率两个方面着手。

在优化布局效果方面，Davidso[22]等人提出的 DH 算法是在能量模型布局算法的基础上进行改进，根据图布局美学标准提出各种约束条件，例如：相邻节点位置接近，减少变交叉等，通过调整这些约束条件的权重使能量函数达到最小值，且布局符合各种美学标准。在提高算法效率方面，各国学者提出了 GEM[23]算法、多尺度[24]算法以及高维嵌入[25]算法。这些算法能够有效提高布局效率，但是却牺牲了一些代价。例如，多尺度算法存在网络直径退化问题，GEM 算法的健壮性差，高维嵌入算法中部分节点存在交叠。

所有这些算法对于一些小图会有比较好的效果，但是随着图的规模的扩大，算法的扩展性不是很好，大图常常会导致能量难以达到最小，当循环次数多时，时间复杂度还有可能会达到$O(n^3)$甚至更高。

3　关联关系可视化布局算法实现

3.1　威胁情报关联关系可视化需求分析

威胁情报的可视化能够让安全分析人员发现未知线索、规律以及异常情况[3].当用户试图分析一个可疑事件时，威胁情报可视化能够帮助分析人员直观地判定可疑事件的恶意性并提供有价值的参考信息，比如通过热力图可以直观地表示事件所涉及的域名是否被已知的 APT 活动使用，通过关系图可以了解相关的 IP 是否在某些已知的黑名单中。这样准确及时的入侵指示数据结合可视化展示效果能够帮助用户快速处理已经发生或正在发生的威胁[26]。

目前，在威胁情报关联关系可视化方面主要存在以下几个问题：

（1）缺乏能够自动化追踪溯源的安全分析技术。

（2）威胁情报关联关系可视化维度单一，拓展性不足。

（3）威胁情报系统关联查询和展示的用户交互体验差。

（4）威胁情报多维、多源异构数据的不合理布局而导致信息混乱，不利于分析。

在工业界，第一章 1.3 相关工作列举了我国威胁情报厂商在关联关系可视化方面的研究成果，ThreatBook、ThreatCrowd、天际友盟三家厂商都使用 Cytoscape.js 这个开源图形库实现威胁情报数据节点关联关系的可视化，但普遍存在交互性不够丰富有效、节点布局效果差、节点无法逐层关联拓展的弊端。

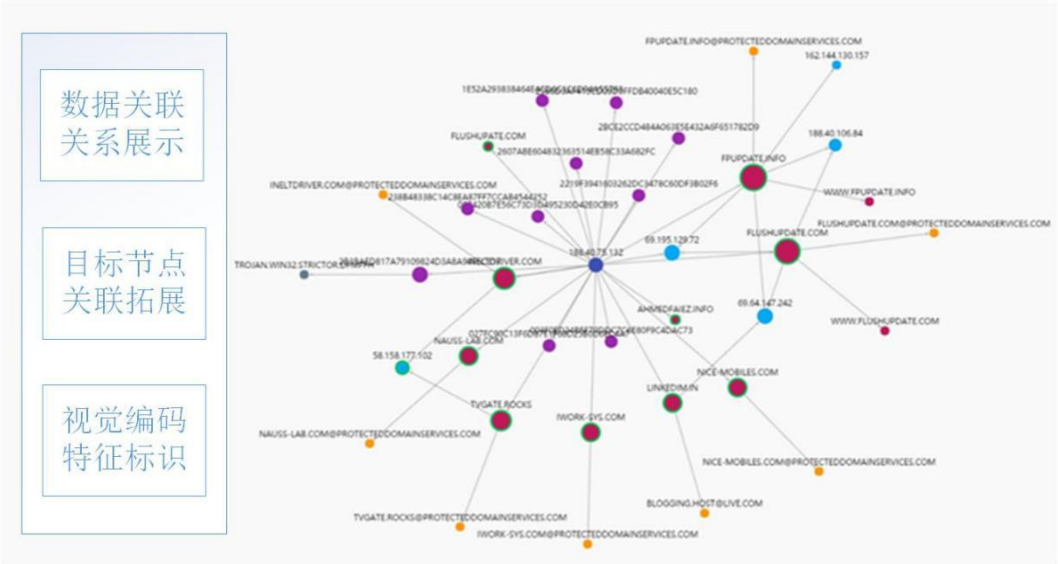

图 3.1　ThreatCrowd 关联关系可视化三大特点

其中，ThreatCrowd 在关联关系可视化方面展示更加丰富，涵盖其他厂商的可视化展示特点，因此，以 ThreatCrowd 为例做现阶段威胁情报产品可视化展示特点分析，如上图 3.1 所示，ThreatCrowd 的可视化界面具有以下三个特征：

（1）数据关联关系展示：通过力引导布局图实现数据节点之间的多层关联关系展示，边表示节点之间的关联关系。

（2）目标节点关联拓展：支持节点的点击扩展，新的节点跳转页面进行新的关联。

（3）视觉编码特征标识：类型不同的节点采用不同的颜色编码标识，节点的大小表示数据的权重，即节点的关联数目。

目前的威胁情报厂商对于关联关系的展示主要包含以上三个特征，但是在用户的交互体验及节点信息的获取中，仍然存在以下几点不足：

（1）数据的再扩展存在断层，对目标节点做关联钻取，会跳转到新页面，以目标节点作为起始节点，之前的关联布局不再保留。

（2）可视化展示静态历史数据，缺乏多源异构数据的实时动态更新机制。

（3）数据量过多时，数据的节点布局与展示效果存在问题，需要优化。

针对上述问题，本文提出针对威胁情报数据关联关系的交互式关联钻取可视化方法，该方法满足目标数据节点在多层关联扩展时保留原始扩展路径，能够动态载入新增数据，并且通过不断地优化节点布局算法能够提升空间利用率，以此来提升安全分析人员的分析效率。

3.2 威胁情报关联关系可视化设计

3.2.1 威胁情报关联关系可视化框架

威胁情报关联关系可视化框架设计需要实现以下几个操作：

（1）数据抓取：通过网络爬虫技术从数十个国内外知名威胁情报厂商处抓取数据，称为外链库。同时，整合以往的数据积累，建立自己的本地数据库，称为核心库。

（2）数据存储：威胁情报关联关系数据存储在图数据库 Neo4j 中。

（3）数据读取：根据用户的需求向后台发送查询条件，从 Neo4j 数据库中读取相关数据。

（4）图形控件生成：根据后台返回的数据信息，将各个节点和边封装成控件，每个控件包含自己的属性，在本文中采用 D3.js 绘制对应的点和边。

（5）可视化布局算法：利用力引导-退火算法对节点进行布局。

（6）交互操作：用户对节点和边进行各种操作，以此触发相对应的监听事件，事件产生的结果通过 WebSocket 协议与 Neo4j 进行交互。

根据上述几个需要实现的操作流程，本文设计了图 3.2 所示的威胁情报关联关系可视化框架，该框架分为两个模块，第一个模块是关联库模块，即后台管理，主要作用在于存储图数据和提供接口；第二个模块是可视化模块，即前端界面，主要的作用是对可视化图形调用图布局算法并且封装交互操作事件。

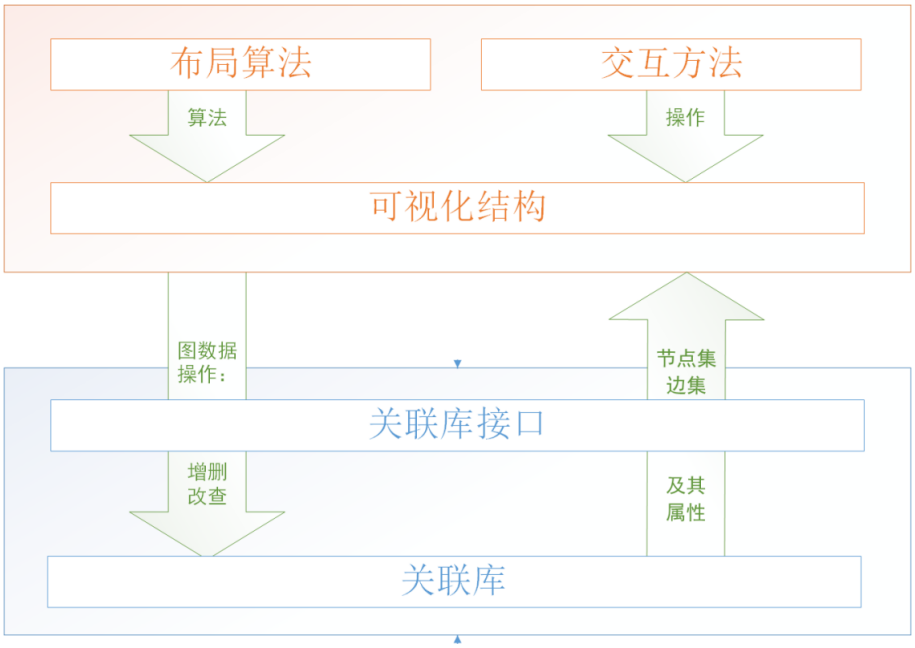

图 3.2　威胁情报关联关系可视化框架设计

本文采用 Neo4j 图数据库来存储和管理威胁情报关联关系数据。用户通过前端可视化界面进行人机交互操作，触发各种监听事件，通过 WebSocket 协议向后台发送数据查询或者关联分析请求，后台图数据库根据查询条件进行检索，通过 WevSocket 协议自发地分批次地向前台返回即时搜索的数据。前端将数据放入缓冲池，调用力引导-退火算法进行动态布局，并即时更新每个节点和每条边的属性，将数据以图的形式展现给用户，以此实现分析的可视化。

3.2.2　威胁情报数据读取

（1）　数据清洗

是指对来源于多个外链库和核心库的数据进行审查和校验，将数据转变为“干净”的数据。该阶段属于数据预处理阶段，其工作包含：

缺失值清洗：针对威胁情报已有数据计算其数据的缺失比例及缺失字段的重要性，分别制定不同策略，如图 3.3 所示。

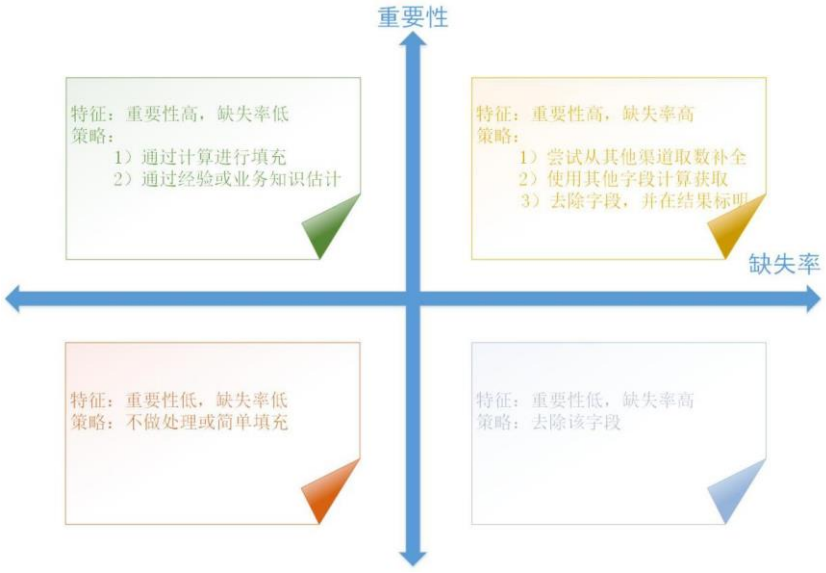

图 3.3　不同特征的缺失值采用不同清洗策略图

去除无效数值：威胁情报数据多维，对于具有弱关联性的属性可以选择去除，在数据去除之前对数据库做好备份工作。删除无效数值，如 IP 的数值不符合 IP 的设置规则，那么该 IP 被判定为无效数值并删除[28]。

格式内容清洗：威胁情报数据来自 virustotal、SHODAN、xForceCracks、站长之家、censys、MAXMIND、NETCRAET、ThreatCrowd、ZoomEye、ThreatBook 等十个外链库和自己的核心库，数据来源丰富、格式多样化、可信度参差不齐，需要研究算法对数据进行融合和格式化，例如查询 ip>220.181.132.217 的住址信息，ThreatBook 采用中文表示，返回北京；ThreatCrowd 采用汉语拼音 beijing 表示；virustotal 采用经纬度表示，返回<39.9289,116.3883>。

（2） 格式转换


```
{
    "nodeClass":"IP",
    "name":"69.195.129.70",
    "id":"69.195.129.70",
    "info":{
        "Address":"United States||Kansas City||39.1068||-94.566",
        "Tag":"'suspicious', 'rdata'",
        "Credibility":"20"
}
```


图 3.4 威胁情报可视化系统后台向浏览器返回的实际数据

存储在数据库中的数据并不能够直接进行可视化展示，需通过处理程序将数据库中已经清洗过的数据转换成 JSON 格式，JSON 格式的优点在于数据占据空间小、数据格式简单易读、便于前台程序处理。如图 3.4 是经过清洗后返回的 JSON 格式数据对象，可以清晰地体现节点类型、节点值、节点基本信息。

威胁情报厂商针对关联关系展示采用力引导-退火算法，针对 IP、DNS、URL 以及 MD5 等类型数据做关联性分析展示。如表 3.1 所示，每种类型的节点都包含不同的基本信息。

表 3.1 威胁情报各种类型数据包含的基本信息

节点类型	IP	DNS	URL	MD5
节点基本信息	所属机构	注册人	存货性	文件名
	互联网提供商	注册地点	行为标签	样本分类
	信誉度	注册时间	初次发现时间	归属家族
	操作系统表示	ISP	情报来源	威胁类型
	涉及协议或应用	子域名信息	关联 Webshell	威胁度
	开放端口	历史解析	Webshell 交互方法	样本链接
	反向 DNS 解析	Whois 信息	Webshell 文件	情报来源
	关联样本	关联样本		来源可靠性
	关联 URL	样本分类		
		样本所属家族		

3.2.3 威胁情报关联关系可视化流程

如图 3.5 所示，威胁情报关联关系可视化流程包含数据读取、可视化控件创建、可视化算法布局、可视化展示这五大模块。

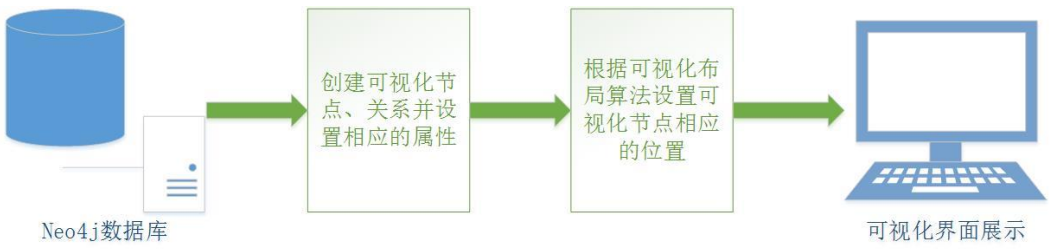

图 3.5 威胁情报关联关系可视化流程图

（1）数据读取模块:从 Neo4j 中读取节点、关系及属性等数据，以 JSON 格式传送给前端。

（2）可视化控件创建模块：为了提升用户体验，将关系图的每个节点设置为可视化控件，并赋予属性和方法，用户可以对其进行各种类型的操作，以此进行人机交互。

（3）可视化算布局模块：通过力引导-退火算法计算每个节点的位置，在画布上绘制每个节点和节点之间因关联关系连接而成的边，形成布局图。

（4）可视化界面：显示关联关系布局图形，用户可以直观的获取信息并进行下一步操作。当进行下一层的关联钻取时，钻取节点与关联节点连接形成边控件。

3.3 关系图布局算法实现和优化

3.3.1 力引导模型

力引导算法常用来描述社交网络等关系型数据，对于威胁情报数据特点，力引导算法构建 IP、DNS、URL 以及 MD5 等数据之间的关联关系[27]。力引导算法通过模拟原子或者天体运动，考虑原子间存在的引力和斥力作用，合力促使节点运动，在运动过程中加入模拟退火过程，使系统最终达到平衡状态。

假设，对于一个高度为 H，宽度为 W 的画布，任意节点 V 都有如下两个参数：

（1）pos：表示节点的位置信息$< x,y >$;

（2）disp：表示节点所受合力产生的位置偏移量;

其中，算法有如下若干基本定义：

（1）显示区域 area：

$$\mathrm{area} = W \times H \tag{3.1}$$

（2）平衡距离 k：

$$\mathrm{k} = \sqrt{\frac{area}{|v|}} \tag{3.2}$$

其中，|v|表示节点的个数

（3）节点之间的几何距离dist(u,v)：

$$\mathrm{dist}(u,v) = \sqrt{(u.pos_x - v.pos_x)^2 + (u.pos_y - v.pos_y)^2} \tag{3.3}$$

（4）相邻节点(u,v)之间的吸引力$f_a(u,v)$：

$$f_a(u,v) = (dist(u,v))^2 / k \tag{3.4}$$

（5）节点(u,v)之间的排斥力$f_r(u,v)$：

$$f_r(u,v) = k^2 / dist(u,v) \tag{3.5}$$

该算法的伪代码如下图 3.6 所示：

```
for(i=1;i<迭代次数;i++)
{
  //计算排斥力
  foreach(v in V)
  {
    v.disp=0;
    foreach(u in V)
      if(u!=v)
      {
        //|pos|是v与u节点之间的位置差
        |pos|=v.pos-u.pos;
        v.disp=v.disp+(|pos|/dist(u,v))*fr(u,v);
      }
  }
  //计算吸引力
  foreach(e in E)
  {
    //各条边是两个节点u，v的序偶
    |pos|=e.v.pos-e.u.pos;
    e.v.disp=e.v.disp-(|pos|/dist(u,v))*fa(u,v);
    e.u.disp=e.u.disp+(|pos|/dist(u,v))*fa(u,v);
  }
  //对点坐标进行置换，防止置换超出边界
  foreach(v in V)
  {
    //温度t限制最大的置换
    v.pos=v.pos+(v.disp/|v.disp|)*min(v.disp,t);
    //防止置换超出边界
    v.pos.x=max(0,min(W,v.pos.x));
    v.pos.y=max(0,min(H,v.pos.y));
  }
  //减少温度，使布局效果更好
  t=cool(t)
}
```

图 3.6　力引导算法伪代码

力引导算法是一个迭代算法，算法迭代次数的设置至关重要，随着迭代次数的增加，算法的布局效果臻于完美，但时间消耗大。因此，如何设置一个折中且合理地迭代次数，是该算法的关键。该算法在每一次循环迭代过程中都会经历了以下 3 个步骤：

（1）计算所有节点来自相邻节点的引力；

$$F_a = ||x_u - x_v||^2/k \tag{3.6}$$

（2）计算每个节点对之间斥力的大小；

$$F_r = \frac{k^2}{||x_u - x_v||} \tag{3.7}$$

（3）每个节点受到引力和斥力的合力作用而运动，运动的幅度受到某个值的限制，该值称为“温度”，随着迭代次数的增加，“温度”减小到 0 或者某个接近 0 的值，节点最终达到静止状态，系统能量达到平衡，图布局完成。

3.3.2 模拟退火算法

力引导模型算法的弊端在于：对于大数据量的节点展示，力引导算法需要对所有节点的位置进行迭代计算，所有的节点都处于不停地运动之中，可能出现以下两种结果：

（1）算法时间复杂度极高，需要很长的时间，才能达到一个理想的布局效果。

（2）节点受力无法平衡，导致不停地运动，会出现图形的震颤效果。

两种结果都会影响用户对于可视化展示图形的视觉体验，因此对于力引导算法需要做优化，加入退火算法使节点运动的位置以更快的速度进行收敛，使图形在短时间内趋于稳定。

在大部分的关系图布局算法中，都使用模拟退火算法来加速收敛。在布局过程中引入“温度”和“冷却”概念，用于限制节点的最大位移距离，并且最大位移量随着迭代次数的增大而减小。“温度”应该从一个初始值开始逐渐衰退到 0 或者趋近于 0 的某个值。随着温度的逐渐下降，节点移动的速度也越来越慢，布局的效果自然也越来越好。在 FR 算法最初的论文[21]中，提到模拟退火的概念，并没有给出计算方法。

Davidson[22]等人为了更好地绘制关系图，对模拟退火算法进行了详细的讨论，但因其考虑的因素过多，许多已经超出模拟退火本身应考虑的范围，因此本文并没有采用其算法。

本文采用线性退火算法，首先设置一个初始温度 T，同时，T 也表示节点的最大位移量。当初始温度 T 足够高时，节点可以大幅度地移动。随着循环迭代次数的增加，温度 T 线性减小。一般将初始温度 T 也就是节点的初始最大位移量设置为画布的一半。本算法的所使用的函数公式如下：

$$\mathrm{T_{v+1}} = \gamma\, \mathrm{T_v} \tag{3.8}$$

γ 被称为冷却系数，取值在 0.55 到 0.95 之间，通过实际应用，发现当$\gamma = 0.75$时，威胁情报关联关系布局效果最佳。模拟退火算法的终止条件是达到迭代次数，该迭代次

数由冷却系数、初始温度、迭代下界三个参数相互作用而决定。迭代下界是指节点的最小位移量，即初始温度线性减小达到的最小值。在实际应用中，我们发现，要达到理想的布局效果，不同数量级节点的迭代次数不同。随着节点数量级的增大，需要更多的迭代次数才能达到更为理想的布局效果。然而，随着迭代次数的线性增长，时间复杂度的增大。因此，如何在布局效果和时间复杂度之间进行权衡，确定一个合理的迭代次数，是本算法的重中之重。

3.3.3 力引导-退火算法优化

在威胁情报可视化系统关联钻取界面应用力引导-退火算法后，通过实际布局效果，调整模拟退火算法的冷却系数、初始温度以及迭代次数，对执行效果进行了分析。经过大量样本测试发现，在实际布局中，当节点数达到百级以上，容易出现部分节点的无用反复震荡，这主要是由于位移量过大，超过了原本应该移动的距离。因此，对力引导算法进行优化，加入了以下控制震荡代码：

```
// 节点位移量偏大
if(dis>bestDis && attrDis>dis)
{
  attr.X*=dis/attrDis;
  attr.Y*=dis/attrDis;
}
// 节点位移量偏小
if(dis<bestDis && attrDis<dis)
{
  attr.X*=bestDis/dis;
  attr.Y*=bestDis/dis;
}
```

图 3.7 力引导算法控制震荡代码

其中，bestDis 表示节点之间的理想距离，dis 表示相邻节点< u, v >的当前距离，attr 表示相邻节点< u, v >之间的吸引力对节点产生的位移向量，attrDis 表示位移后相邻节点之间的实际距离。

由上述伪代码可以看出，当相邻节点< u, v >的实际距离小于理想距离时，应该增大< u, v >之间的距离。但是，如果节点进行为位移后，< u, v >之间的距离反而减小，这时，就该增大位移量。同理，相邻节点< u, v >的实际距离大于理想距离时，应该减小<

u,v >之间的距离。但是，如果节点进行为位移后，< u,v >之间的距离反而增大，这时，就该减小位移量。

通过大量样本测试表明，增加了控制震荡代码后，无用节点反复震荡的情况出现率大大降低，关系图的布局效果也更加理想。但是，该算法增加了对每组相邻接点距离的计算，这增加了算法的复杂度，但是复杂度的量级并未增大。可以适当降低力引导算法和模拟退火算法的迭代次数，来抵消这一部分时间消耗。

通过 3.3.2 分析表明，引入模拟退火算法可以有效地防止节点位移震荡。因此，在力引导算法的循环迭代中，引入模拟退火线性算法来限制节点位移的最大偏移量，随着循环次数的增加，节点位移的最大偏移量递减，直至减少到 0 或者接近 0 的某个最小偏移量值。伪代码如下：

```
//力引导-退火算法迭代
for(i=0;i<iterationCount;i++)
{
  //引入模拟退火算法
  annealingOffset*=(double)(annealingOffset-k-1)/iterationCount;
  //力引导算法计算
  Computing();
}
```

图 3.8 力引导-退火算法伪代码

其中，iterationCount 表示力引导算法的迭代次数，annealingOffset 表示模拟退火算法中节点位移的最大偏移量，也表示初始温度 T。实验表明，在力引导布局中合理引入模拟退火算法可以减少时间复杂度和实现更好的布局效果。

同时，在威胁情报关联钻取可视化布局算法中，尝试采用网格变量方法对计算排斥力的步骤进行了优化。具体内容如下：将画布中节点布局的区域分为若干个网络，计算排斥力时，不再计算节点 v 与所有其它节点的作用，而是只考虑节点与相邻网格内的其他节点之间的作用。该算法的排斥力计算公式如下：

$$f_r = \frac{R^2}{d} u(2R - d) \tag{3.9}$$

其中，R 表示空白区域的理想半径，d 表示两个节点之间的距离。U 的值有如下两种情况

$$\mathrm{u(x)} = \begin{cases} 1, & if\ x > 0 \\ 0, & otherwise \end{cases} \tag{3.10}$$

3.4 数据动态加载设计与实现

本论文在第四章 3.2.2 提到威胁情报关联钻取的数据来源于 censys、virustotal、SHODAN、xForceCracks、站长之家、MAXMIND、NETCRAET、ThreatCrowd、ZoomEye、ThreatBook 等十余个外链库和自己的核心库。在传统的 HTTP 模式下，当安全分析人员在前端输入查询条件或者节点拓展条件时，浏览器对服务器发送 HTTP 请求，服务器端需要对十几个外链库和核心库进行查询，将查询的结果返回给客户端的浏览器。这会导致两个问题：

（1）对十余个库进行查询的时间太过漫长，严重消耗用户的耐心，不利于用户交互；

（2）HTTP 只能实现浏览器对服务器的单向通信，也就是说，当服务端获取新的数据时，并不能及时返回给客户端，这会导致前端可视化界面仍显示历史关联数据，对于数据节点的关联钻取和追踪溯源分析的准确性严重降低。

因此，本小节通过实现威胁情报关联钻取可视化的新增异构数据异步动态加载功能，来解决上述不足。

WebSocket 是一种基于 TCP 的网络新型协议，实现了前后端的全双工通信，即服务器可以主动发送信息给客户端。其业务模型如下图 3.9 所示：

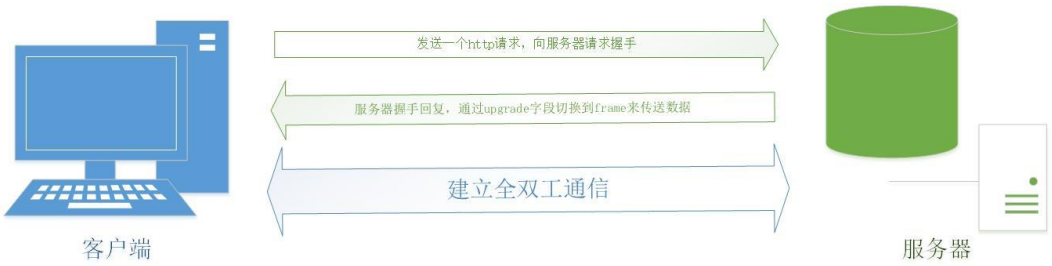

图 3.9 Websocket 业务模型图

在实现 WebSocket 的连线过程中，浏览器向服务器发送 HTTP 请求，请求建立 WebSocket 连接，浏览器对此发出回应，这个过程通常称为“握手”。仅需一个握手操作，浏览器和服务器就形成了全双工通道，两者之间可以互相传送数据，最有出彩的特征是服务器不再被动的接受到浏览器的 request 请求后才返回数据，而是在有新数据时就主动推送给浏览器。该协议在本系统实际应用的函数定义代码如下图 3.10 所示。

威胁情报数据的动态加载能够实现新增数据从后台到前端的及时推送，包含两个应用情景：

（1）用户做关联分析对节点进行关联扩展。

```
2067    //start-此函数用于实现浏览器与服务器的全双工通信
2068    function websocket_asso(url, params, successHandle, cancleAll) {
2069      var ws = null;
2070      if (window.WebSocket) {
2071        ws = new WebSocket(url);
2072
2073        ws.onopen = function () {
2074          console.log("connect OK");
2075          ws.send(JSON.stringify(params));
2076        };
2077        ws.onerror = function () {
2078          console.log("connect ERROR");
2079        };
2080        ws.onclose = function (e) {
2081          cancleAll();
2082          console.log(e);
2083          console.log("connect CLOSE");
2084        };
2085        ws.onmessage = function (msg) {
2086          successHandle(msg.data, this);
2087          // console.log(msg.data);
2088        };
2089      }
2090      return ws;
2091    }
2092    //end-此函数用于实现浏览器与服务器的全双工通信
```

图 3.10 websocket 函数定义代码

对于威胁情报数据的每个节点，都可以进行逐层关联钻取，本系统实现了关联全部、关联 MD5、关联 URL、关联 DNS、关联 IP、关联事件、关联组织、关联邮箱等八种关联功能，实际预览图如图 3.11 所示。

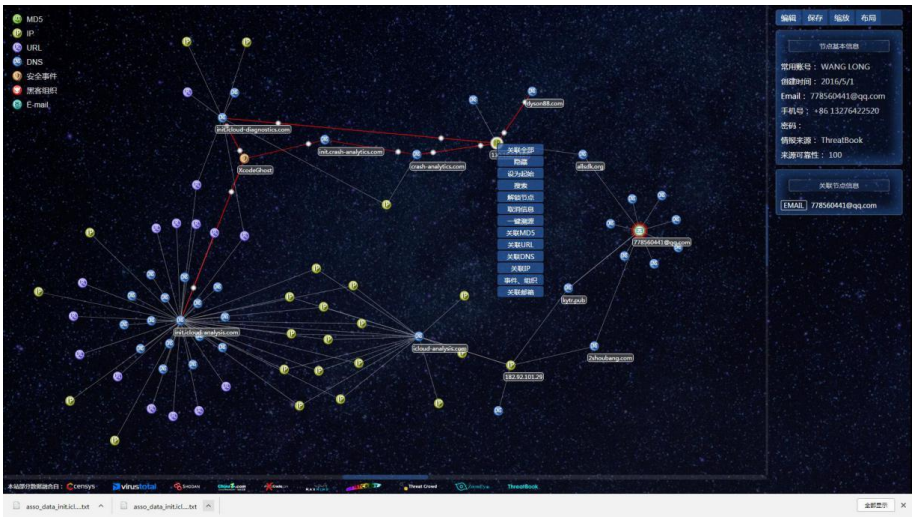

图 3.11 威胁情报可视化系统关联钻取实际预览图

当用户期望对某个节点继续探索关联关系，右键点击该节点，选择要关联的数据类型，图形会继续扩展目标节点的关联数据，并以力引导-退火算法进行布局，其流程如下图 3.12 所示。

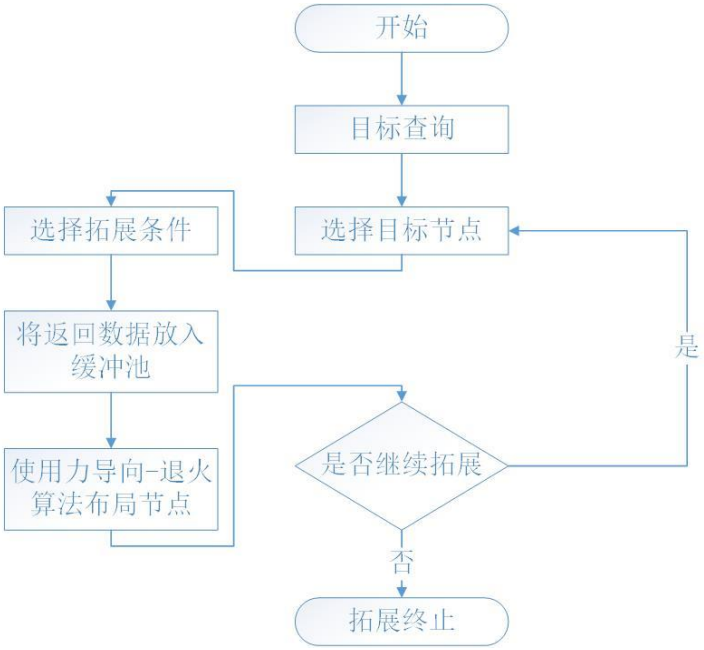

图 3.12 节点关联拓展流程图

（2）新增数据的自动关联展示

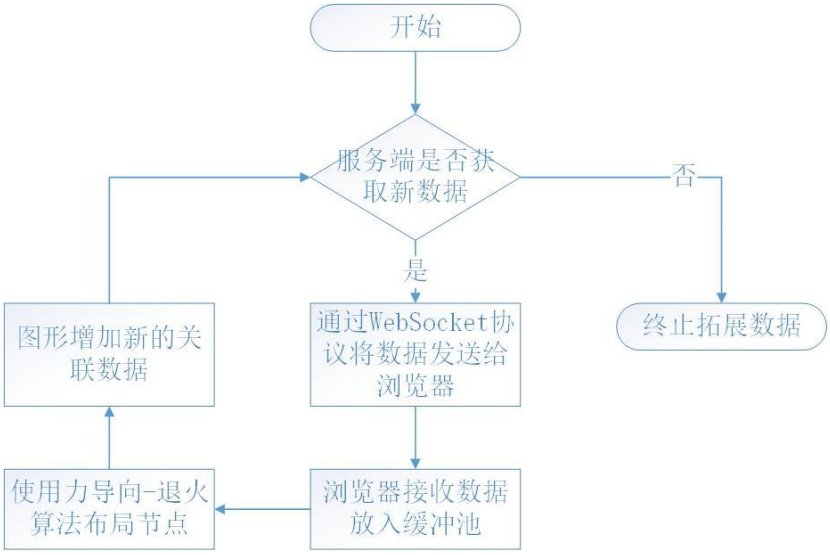

图 3.13 新增数据的自动关联展示流程图

在本可视化系统中，由后台支撑对威胁情报平台的数据搜索，后台数据库不断地更新。在安全分析人员进行关联钻取时，通过 WebSocket 技术，后台主动向前台实时推送新增异构数据，这样不仅增加了关联钻取数据的准确性，同时优化了用户体验，其流程如上图 3.13 所示。

4　威胁情报可视化系统设计

4.1　威胁情报可视化系统功能需求分析

功能需求是指软件开发者应为用户提供特定的功能和服务供其使用，满足用户的正常需求。通过对安全分析人员的用户需求进行调研，本论文设计的威胁情报可视化系统包括以下几个功能模块：用户登录模块、情报查询模块、可视分析模块、情报维护模块、情报统计模块。具体的功能需求如图 4.1 用例图所示。

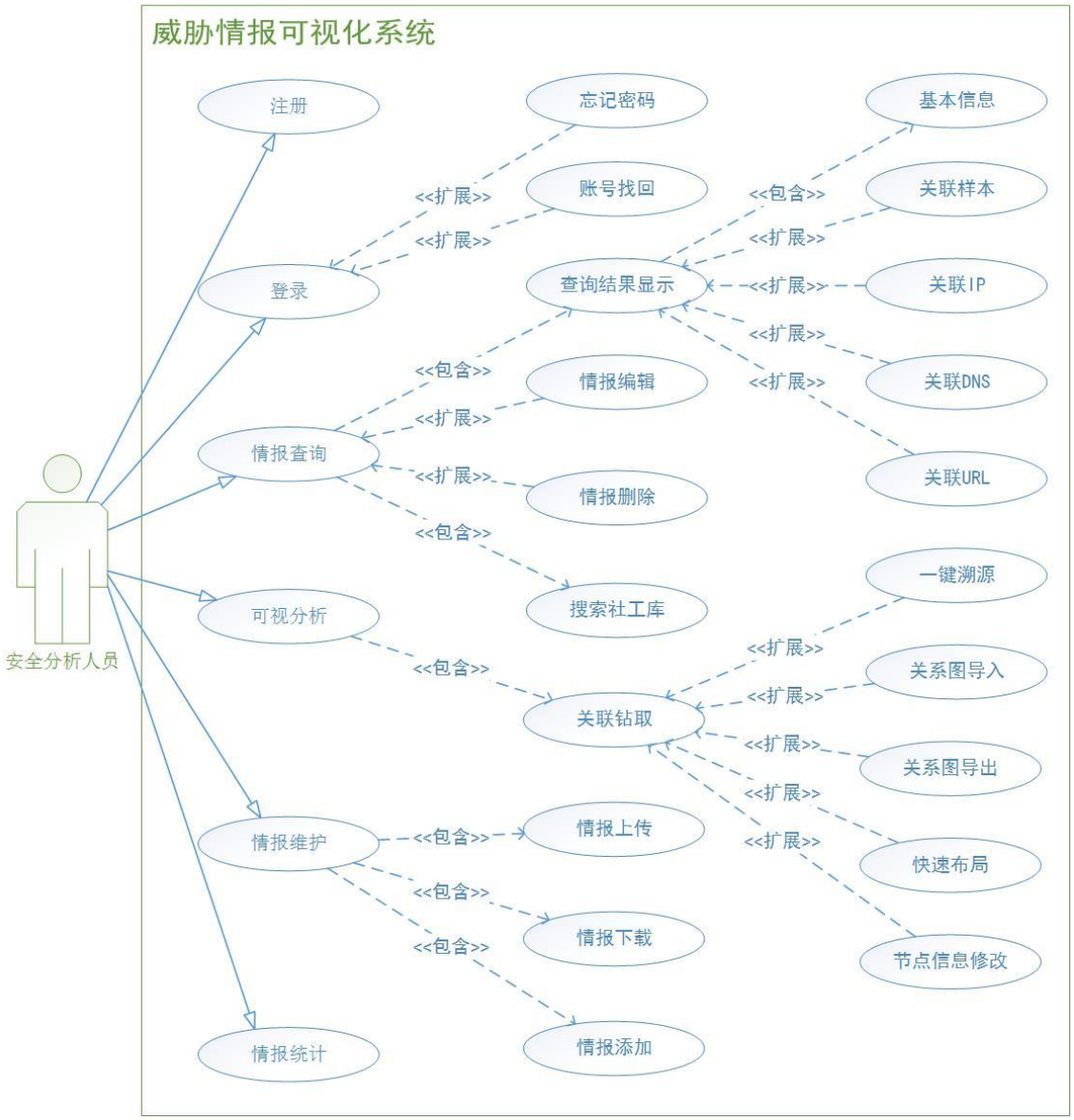

图 4.1 威胁情报可视化系统用例图

4.2 威胁情报可视化系统框架设计

4.2.1 设计原则

本文结合网络安全行业的实际特点，以及归纳总结安全分析人员对系统的实际使用需求，在设计威胁情报可视化系统框架时，应遵循以下五大原则：

（1）安全性原则

威胁情报可视化系统属于网络安全产品。首先，应该保障自身系统的安全性，即该系统的各个模块之间的通信应有严格的防窃听措施。同时，数据存储和告警信息要保证使用单位的隐私不泄密。

（2）松耦合性原则

威胁情报可视化系统采用 vue.js 架构，使用单文件组件模式进行开发，坚持“一个模块，一个功能”的原则，功能模块之间应该尽量减少耦合，使系统具有良好的重用性和维护性。

（3）可拓展性原则

威胁情报行业发展日新月异，为适应网络安全防护业务需求变化快的特点，系统在设计时应合理架构，保证快速迭代开发。系统在设计和测试直到最终交付时，都要以用户为中心，UI/UE 设计要人性化，做到真正符合最终用户的实际需要。

（4）稳定性原则

系统的前后端架构都要使用成熟的技术，硬件基础设施要完备，以保证系统实际运行的稳定性。

（5）可维护性原则

系统具备优质的管理、监控、故障分析和处理能力，可集中统一维护，在功能划分和设计时，使各模块尽可能相对独立、减少耦合性，便于维护。

4.2.2 平台架构设计

本论文基于威胁情报可视化统，系统架构分为三部分：浏览器端、服务器端以及数据库端，具体架构图如图 4.2 所示。

数据库端使用 HBase、ES、Neo4j 三类数据库，分别存储不同类型的数据。HBase 是一个高性能、面向列的分布式存储系统，适合威胁情报多维、多源、异构基础数据的存储。Elasticsearch（简称 ES）是一个兼有搜索引擎和 NoSQL 数据库功能的开源系统，是当今最为先进和高效的全功能开源搜索引擎框架。实验室爬取大量的社工库数据，将他们存储于 ES 中，并为他们建立索引，这样就能实现社工库的快速搜索。Neo4j 是一个高性能的 NoSQL 图形数据库，它是一个面向对象的、灵活的网络结构而非严格、静态的

表，善于处理大量复杂的、互相连接、低结构化的数据，这些数据变化迅速、需要频繁的查询，本文中使用的威胁情报关联数据正是存储于 Neo4j 中。

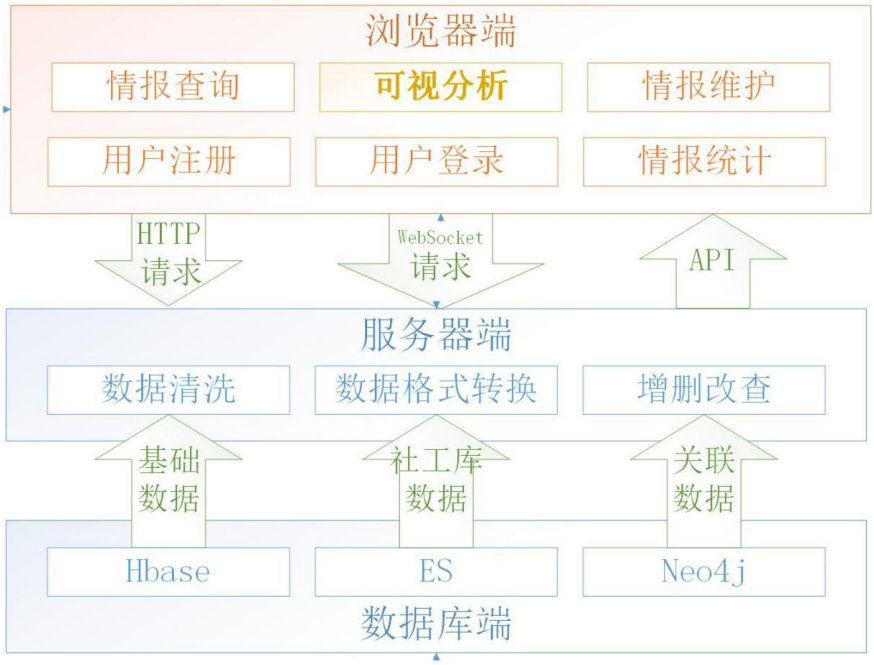

图 4.2　威胁情报可视化系统架构图

服务器端开发主要是进行数据处理和封装请求 API。由上文可知，本系统威胁情报数据来源于国内外数十家知名威胁情报厂商和实验室内部积累的数据。这些数据重复率高、可信度低、格式参差不齐，在使用前需要对数据进行清洗、格式转换、可行度排序。

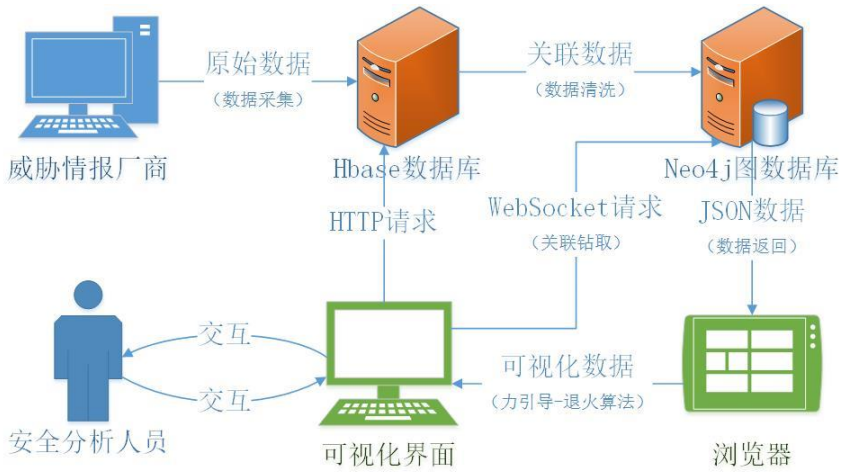

图 4.3 威胁情报可视化框架数据流概要图

浏览器端负责前端系统的开发架构，使用 vue.js 搭建数据驱动的 web 界面渐进式框架，vuex 做状态管理，vue-router 做路径切换，vue-resource 做数据通信，WebSocket 做全双工通信，D3.js 实现可视化展示与交互，element-UI 实现组件快速开发。

本论文的重中之重在于威胁情报关联关系可视化设计，从可视化角度来看，系统分为三个模块：数据采集模块，数据处理模块以及可视化展示模块。数据采集模块将实验室内及威胁情报厂商公开的资源通过数据融合处理后存储到数据库中，通过数据处理技术将数据库中有效数据进行清洗，并转换成符合前端标准的 JSON 格式数据，前端将数据放入缓冲池，通过力引导-退火布局算法对节点进行布局，向用户展示可视化界面。该描述的数据流图如图 4.3 所示。

4.3 使用技术介绍

本论文所依托的威胁情报可视化系统前端使用 vue.js 搭建数据驱动的 web 界面渐进式框架，vuex 做状态管理，vue-router 做路径切换，vue-resource 做数据通信，WebSocket 做全双工通信，D3.js 实现可视化展示与交互，element-UI 实现组件快速开发。以下篇幅对各项技术做详细介绍。

4.3.1 Vue.js

Vue.js 是一个构建数据驱动的 web 界面的渐进式框架，目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。Vue.js 是当前业内最好用的 MVVM 框架，可以拆分成 View-ViewModel-Model 三部分，具有轻量级、上手快的优点。Vue.js 包含以下两大核心：

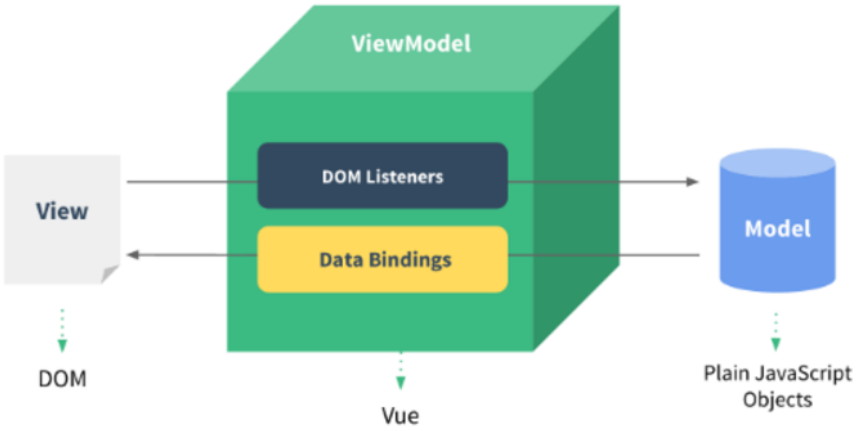

图 4.4 vue.js 的 MVVM 架构模型图

（1）数据驱动：传统的实现页面更新的做法是手动改变 DOM 来更新视图，而 Vue.js 只需要改变数据，页面就会自动做相应的变化，这称为数据驱动，也是 MVVM 思想的实现。举个例子，当我们想要通过点击一个 button 来切换某个 div 标签里面的文本时，需要对 button 绑定事件，通过该事件手动获取 div 对象的文本值，并且通过 toggle 来切换该文本值。而使用 vue.js 实现这个功能，只需要在 button 上指明事件，同时声明对应 div 标签的属性，点击 button 时就可以改变属性，对应的文本也能进行自动切换。Vue.js 的 MVVM 架构如下图 4.4 所示，vue.js 封装了数据和 DOM 对象操作的映射，viewModel 监听双方的动作，只要数据一发生变化，viewModel 会通知页面进行重新进行渲染。同样，当页面有事件触发时，viewModel 会通知 model 进行响应。

（2）视图组件化：威胁情报可视化系统的开发正是采用单文件组件进行开发，将一个网页拆分成一个个组件，网页有多个组件拼接或者嵌套组成，组件可以复用。如图 4.5 所示是威胁情报系统交互式关联钻取页面，该页面由 LeftNav、HeadNav、Findsrc、Tips、Association、ShowInfo 六个组件构成，每个组件又由若干个子组件构成。

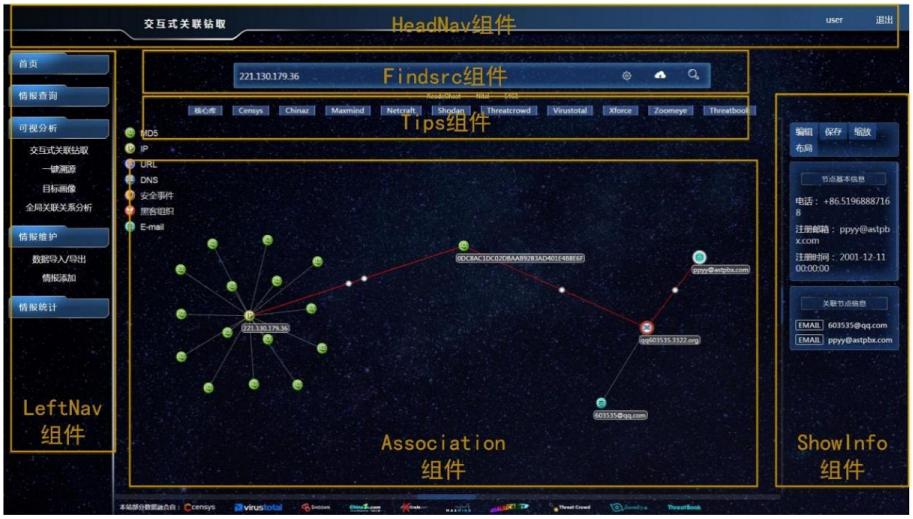

图 4.5 威胁情报关联钻取界面组件构成示意图

本论文所依托的威胁情报可视化系统正式运用 Vue 驱动采用单文件组件和 Vue 生态系统支持的库开发的复杂单页应用。

4.3.2 Vuex

Vuex 是专门为 Vue.js 应用设计的一款集中式状态管理模型库。当我们使用 Vue.js 时，通常会把状态保存在组件的内部，换句话说，每一个组件都拥有整个应用状态的一部分，整个应用的状态分散在各个组件中。在威胁情报可视化系统的开发中，经常需要

把应用状态的一部分共享给多个组件，例如，用户信息。然而在多层嵌套的组件中进行传参是十分繁琐复杂的，而且当组件想要变更应用状态时需要同步状态的多份拷贝，这种开发模式是十分脆弱的，极其容易导致代码无法维护。因此，本系统引入 Vuex 实现共享状态管理，即将多个组件中的共享状态提取出来，作为全局单件进行管理，这样可使代码更结构化和具有高可维护性。

4.3.3 Vue-router

Vue-router 是与 Vue.js 深度集成的路由插件。在传统的页面应用中，一般用超链接来实现页面的切换和跳转功能。在威胁情报可视化系统的各个单页面应用中，通过 Vue-router 设定访问路径，并将路径和组件映射起来，实现路径之间的切换，也就是组件之间的切换。如图 4.6 是可视化系统路由切换的部分代码，25 行的 content 组件表示顶部导航栏和侧面导航栏以及整个页面框架，26 行折叠的 children 组件表示情报查询、关联钻取、一键溯源、情报导入/导出、情报添加、情报统计等各个子组件，通过路由切换，实现各个子组件的切换。

```
15  Vue.use(VueRouter);
16  let route = [
17      {
18          path: '/index',
19          name: '首页',
20          component: search
21      },
22      {
23          path: '/content',
24          name: 'content',
25          component: content,
26 ›        children:[…
113     }
114 ];
115 route.push({ path: '*', redirect: '/index' });
116 const router = new VueRouter({
117     mode: 'hash',
118     /* default */
119     routes: route,
120 });
```

图 4.6　威胁情报可视化系统路由切换源代码

4.3.4 Vue-resource

Vue-resource 是一个通过 XMLHttpRequrest 或 JSONP 技术实现异步加载服务端数据的轻量级的 Vue 插件。Vue-resource 相较于 jQuery 来说，体积非常小巧，使用方便，本系统的开发正是使用 Vue-resource 进行 http 请求，实现威胁情报数据的增删改查等功能。

4.3.5 WebSocket

论文第三章 3.4 详细描述了使用 Websocket 技术实现浏览器与服务端的全双工通信，此处不再累赘。

4.3.6 D3.js

D3.js 是一个基于数据驱动 DOM 的 JavaScript 库，使用 HTML、SVG 和 CSS 来实现强大的可视化功能，让用户以数据驱动的方式来操作 DOM。例如，使用 D3.js 可以将一个一维数组生成为 HTML 表格，或者用其创建一个带有平滑过渡和互动功能的交互式 SVG 柱状度。本论文所依托的威胁情报可视化系统的交互式关联钻取功能、一键溯源功能、情报统计功能正是使用 d3.js 进行开发。下图 4.7 是系统用 d3.js 实现的可视化图形的预览图。

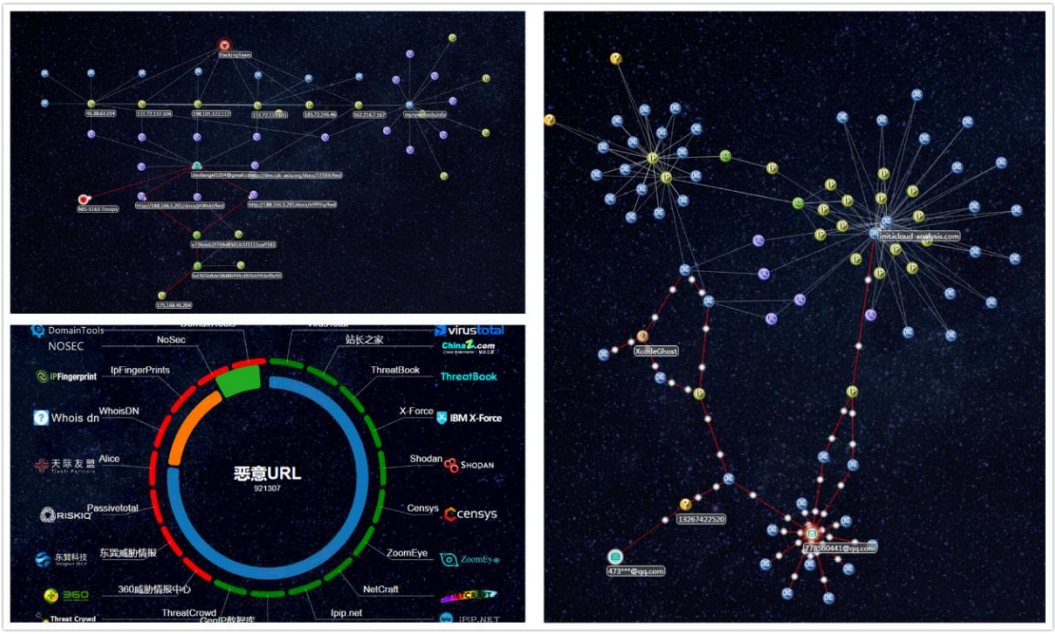

图 4.7　威胁情报可视化系统关联钻取与情报统计实际预览图

4.3.7 Element-UI

Element-UI 一套基于 Vue 2.0 开发的组件库，能够提供配套的设计资源，可以帮助网站快速成型。在本系统的开发中，通过对 Element-UI 源码进行修改和重新编译，使组件样式与系统设计风格相匹配，将其应用到情报添加、编辑节点的功能中，实现组件的快速开发。下图 4.8 是威胁情报可视化系统节点编辑功能的实际预览图，该图中编辑节点的模态框就是使用 Element-UI 进行快速开发的。

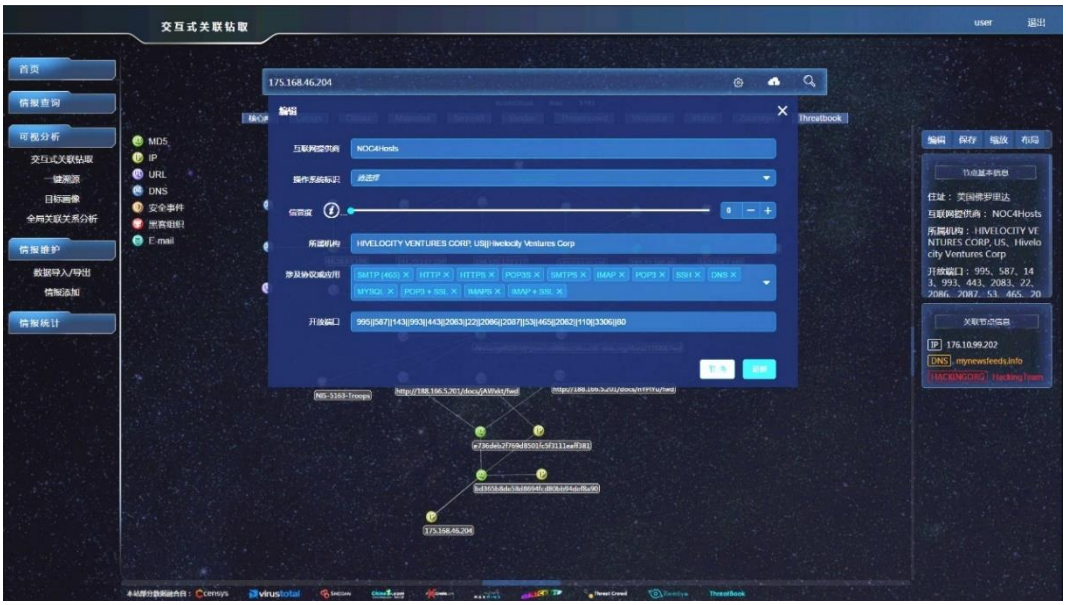

图 4.8 威胁情报可视化系统节点编辑功能实际预览图

4.3.8 Vue-cli

当使用 vue.js 进行开发时，或许有人认为，所需要做的仅仅是在每个页面通过<script>标签引入 vue.js。但实际并非如此，在真正的项目开发中，需要使用模块化、预处理器、热模块加载、代码校验和测试模块等一系列工具，这些工具的初始化并不简单。而 vue-cli 能自动帮助开发者完成这些配置，它是 vue.js 的脚手架，通过一个简单的命令行，能够快速构建一个强大的 vue.js 模板工程。

5　威胁情报可视化系统实现与测试

5.1　用户登录模块

对于威胁情报可视化系统这类安全产品，必须要有用户权限才能进行操作。因此，登陆功能是系统必不可少的功能之一。同时登陆界面提供密码找回功能，方便用户及时找回密码。如图 5.1 是登陆模块的流程图。

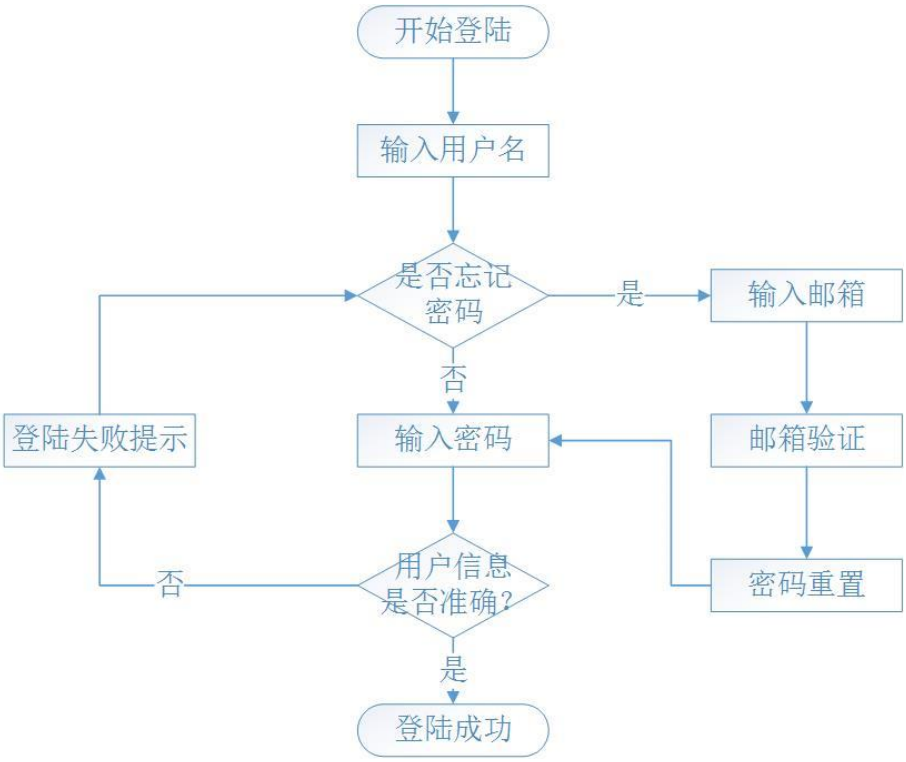

图 5.1　登陆模块流程图

考虑到安全性问题，进行登陆操作时，前端对用户名+IP+用户类型进行 Base64 编码，对密码进行 md5 算法加密操作，并存入 Cookies。在判断用户是否登陆时，检验相应的 Cookies，对编码进行 Base64 解密操作，获取用户名。从数据库获取密码进行 md5 加密，然后比较 IP 地址，进行密码校对。如若信息匹配，则登录成功，服务器根据用户权限级别返回相应的功能开放权限数组，前端通过数组值，展示相对应的功能模块，对于权限不够的功能模块，前端进行路由禁用，不予展示。

下图 5.2 为用户登录界面实际预览图，当用户成功登陆时，跳转到主页面。

图 5.2 用户登录界面实际预览图

5.2 情报查询模块

威胁情报查询，是对情报的基本信息和关联数据进行查询，支持的查询类型有 IP、DNS、MD5、URL、样本值。图 5.3 是系统进行威胁情报查询的实际预览图。

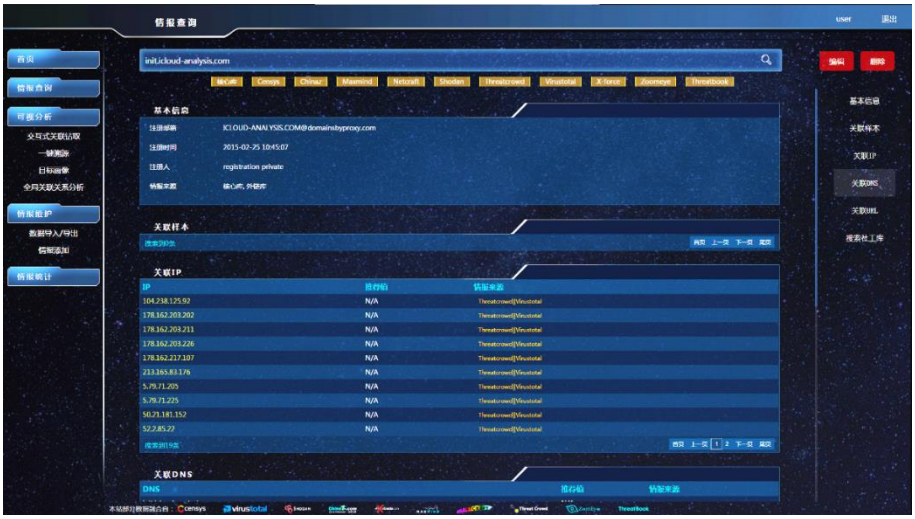

图 5.3 威胁情报可视化系统情报查询实际预览图

图 5.3 是对域名>init.iclout-analysis.com 的搜索结果，包含基本信息、关联样本、关联 IP、关联 DNS、关联 URL 等来源于十几个外链库和核心库的数据。当用户点击搜索图标时，前端向服务器请求建立 Websocket 全双工通信，并将查询参数返回给后端。由于对外联库的查询效率不同，服务器无法一次性返回所有数据，而是通过异步加载的方

式将搜索到的数据不定时的返回给浏览器，因此，前端制作了跑马灯功能，当相应库的数据返回时，高亮显示数据来源名称，如图 5.4 所示。

图 5.4 情报查询跑马灯功能

同时，前端对返回数据的类型进行判断，如果是可进行二次搜索的数据，则高亮显示。当用户点击对应的名称时，触发查询函数，进行二次搜索。例如，点击关联 IP 中的 104.238.125.93，则可对该数据进行搜索。该页面也实现了回退功能，设置一个栈，将每次的查询 id 放入栈中，当用户点击回退按钮时，从栈中取出最顶层的值，触发该值的查询函数，进行搜索。

本系统采用 Vue.js 架构，是以数据驱动的，即根据返回数据量的大小，动态布局每个子组件框。如图 5.3 所示，域名>init.iclout-analysis.com 的关联样本数据为 0 条，因此关联样本子组件表格为空，关联 IP 数据为 19 条，关联 IP 子组件表格进行分页展示。因为关联样本、关联 IP、关联 DNS、关联 URL 的表格样式相同，因此，它们复用同一个组件，根据返回数据格式的不同，显示不同数据，这就是 MVVM 架构的魅力。

5.3 可视分析模块

5.3.1 关联钻取功能实现

关联钻取是本系统重点实现的功能，也是开发量最大的功能模块。在本论文第三章已经详细阐述了关联关系可视化布局算法，以下篇幅主要从功能实现和用户交互方面进行阐述。

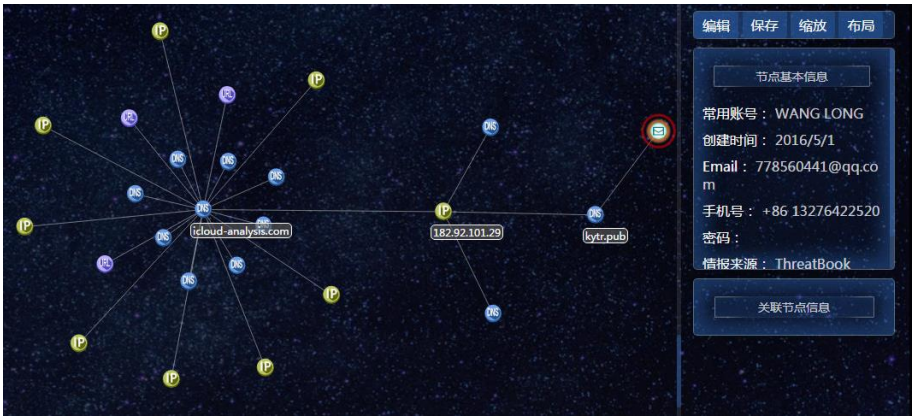

图 5.5 icloud-analysis.com 关联钻取预览图

下面以对域名>icloud-analysis.com 进行关联钻取为例，阐述本部分功能。实际预览图如图 5.5 所示。图 5.5 的关联钻取流程如下：

（1）在搜索栏中输入“icloud-analysis.com”，并选取“关联全部”，实现对节点的全部关联节点扩展。

（2）选取“关联 IP”，钻取出来 “182.92.101.29”。

（3）选取“关联 DNS”，钻取出来“kytr.pub”。

（4）选取“关联邮箱”，钻取出来“778560441@qq.com”，点击邮箱节点“778560441@qq.com”，可以看见攻击者的常用账号，Email、手机号等社会信息，达到分析目标终止节点扩展。

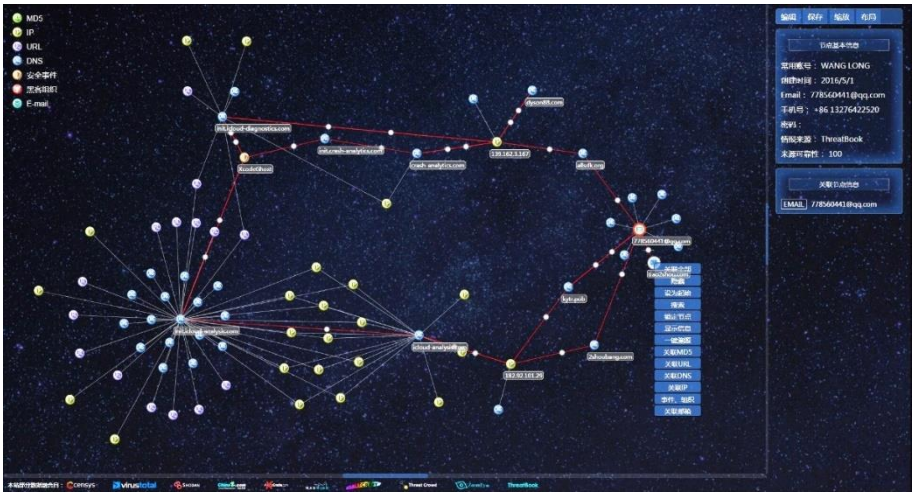

图 5.6　xcodeghost 事件关联钻取预览图

至此，通过对域名>icloud-analysis.com 关联钻取，最终得到攻击者的真实社会信息。当然，安全分析人员如想了解更为复杂的关联关系，可继续向下进行关联钻取。如图 5.6 所示，是对 xcodeghost 溯源分析展示，可以发现该事件关联多个域名，通过钻取，最终溯源到同一个邮箱账户“778560441@qq.com”。xcodeghost 事件包含多条关联路径，其中，图 5.5 对域名>icloud-analysis.com 溯源的流程，就是其中一条路径。

本文采用力引导-退火算法对关联钻取节点进行布局，前端每次接收新数据时，都会调用算法再次布局，每个节点都会受到来自相邻节点的引力和其他节点的斥力作用所产生的合力，沿着合力的方向进行移动，直到达到平衡状态或者最大迭代次数。然而，往往当算法迭代过程还未结束时，用户就进行新的操作，节点受到外力作用而静止，无法达到一个最佳的布局效果。而且，安全分析人员在进行关联操作时，频繁地拖动节点，当溯源结束，整个关联图已面目全非。针对于此，本系统实现了一键布局功能，用户操

作的过程中，可随时对界面已有的节点进行重新布局。其原理是随机选择某个节点为中心，运用力引导-退火算法计算得出每个节点的位置，然后在画布上绘制出该图形。关联钻取页面的一键布局效果如图 5.7 所示。

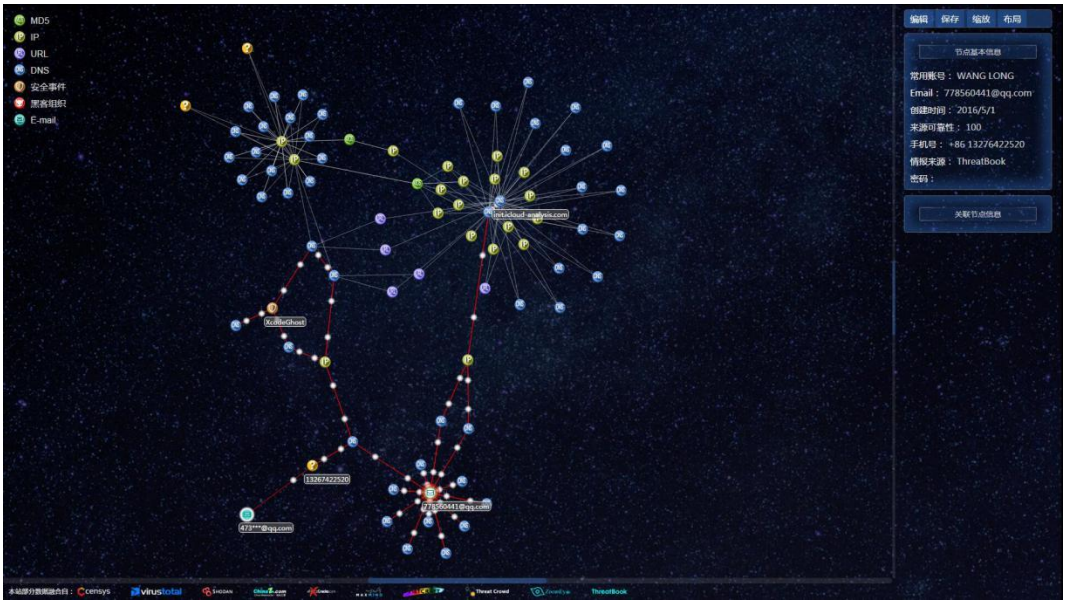

图 5.7 关联钻取一键布局功能效果预览图

当安全分析人员对一个攻击事件进行关联钻取和追踪溯源时，往往并非能够一蹴即成，而且随着情报共享平台的不断完善，威胁情报数据也愈加丰富。因此，事件的关联钻取和追踪溯源可视化图形并非一成不变，如何实现关系图的保存和再操作成为系统开发的一个硬性需求。论文通过编码实现关联钻取布局图的本地保存和导入功能，将关系图的每个节点每条边的属性以 JSON 格式进行保存。


```
"node": [ {
    "id": "node2",
    "name": "198.105.125.107",
    "nodeClass": "IP",
    "occurTime": "",
    "info": {
        "Address": "捷克兹林州"
    },
    "xy": [7371.15537656015, 3893.8496675875886],
    "linkNumber": 2,
    "isChoose": false,
    "noMove": true,
    "lockedInfo": false,
    "isShow": "010000"
}],
```

```
"link": [{
    "source": "node0",
    "target": "node1",
    "sourceId": null,
    "targetId": "bd365b8de58d8694fcd80bb94def8a90",
    "sourceXY": [7080.96875, 4061],
    "targetXY": [7224.572471891897, 3977.383410185173]
}, {
```


图 5.8 关系图保存文件真实数据格式

如上图 5.8 所示，是关系图保存文件节点和边的真实截图。节点保存的信息包括节点编号、节点名、节点类型、生存时间、节点基本信息、节点坐标、连接节点的边数、节点是否被选中、节点是否固定、节点名是否显示、该类型节点是否展示等一系列属性。边保存的信息包括源节点编号、目标节点标号、溯源路径源节点编号、溯源路径目标节点编号、源节点坐标、目标节点坐标等一系列属性。当将文件导入时，调用封装好的绘图函数还原现场，根据节点和边的属性绘制图形，可返回到与保存前相同的状态，用户可在该图形的基础上继续进行操作，同时也可点击更新，当已存在的节点有新的关联信息时，后台通过 WebSocket 协议向前台返回新数据，前台调用力引导-退火算法进行重新布局。

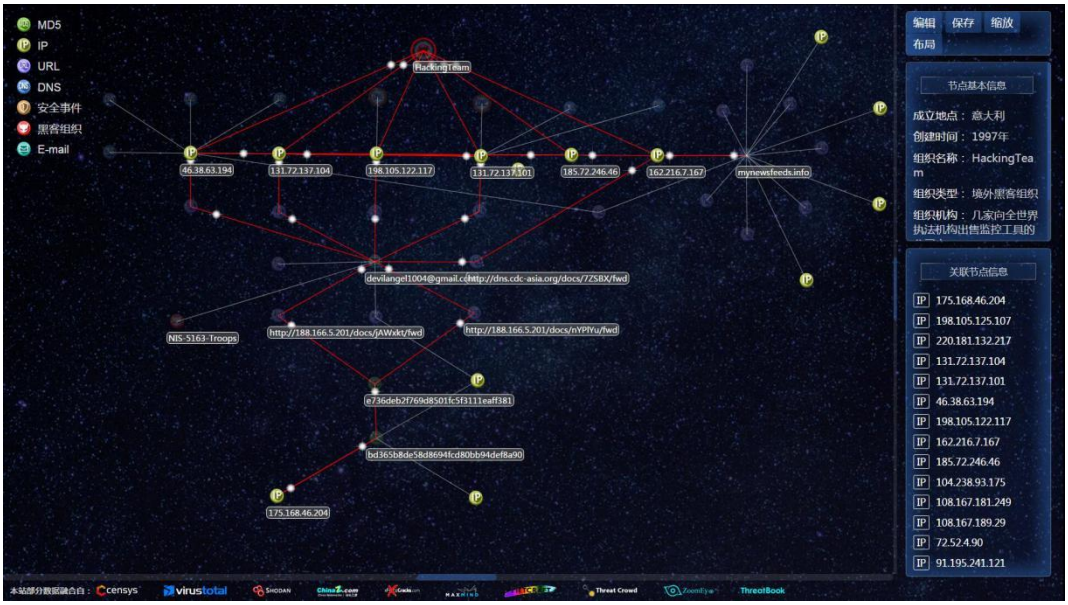

图 5.9　Hacking Team 关联钻取示意图

如图 5.9 展示 Hacking Team 组织的关联关系，是由安全分析人员反复钻取得到的结果，具有典型意义。由于关系图的保存和再操作功能的实现，该图可以放在服务器端作为典型例子，供所有用户浏览参考，同时，用户也可以在此图的基础上进行修改或继续向下钻取。由图中可以看出 IP 类型的节点高亮显示，其余类型的节点半透明显示，这也是系统开发的功能之一。通过点击左上角各种类型的节点，可以高亮显示被选中的类型的节点，同时所有的节点出现在右下角的关联节点信息表格中，通过选取表格中的节点，可以选中图中相关的节点，例如点击关联节点信息表格中的 IP>175.168.46.204，关系图中的该节点就会被选中，用户可对该节点进行下一步操作。

5.3.2 一件溯源功能实现

本系统关联钻取节点信息存储于 Neo4j 数据库，通过后台编程，对图数据库进行检索，可以获取威胁情报节点关键路径数据，实现一键溯源功能。一件溯源功能的实现细节同关联钻取一样，采用力引导-退火算法进行布局，唯一的区别是数据源的不同。

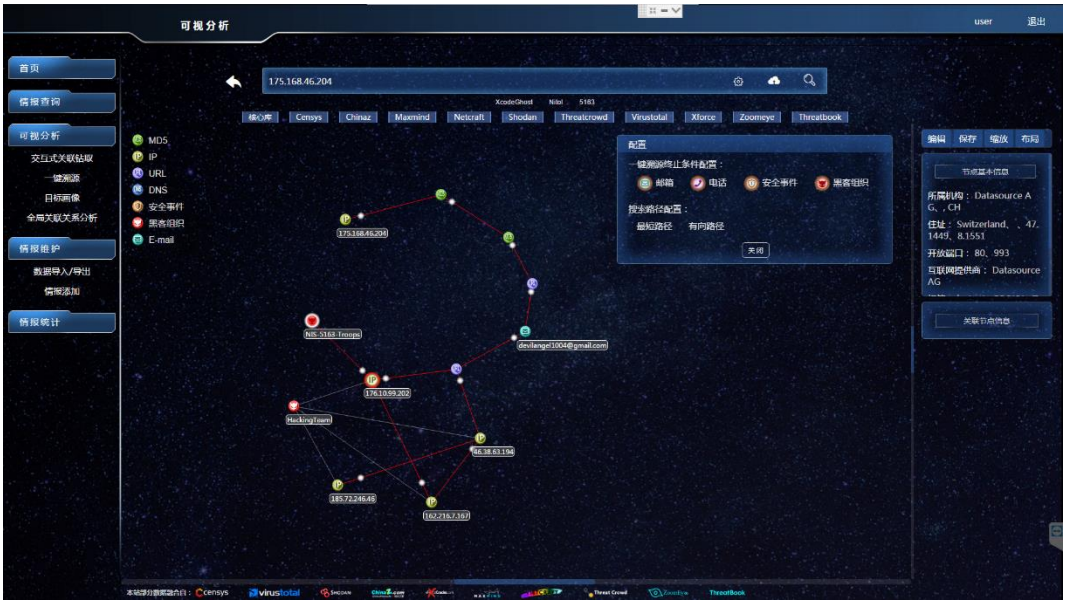

图 5.10 威胁情报可视化系统一键溯源功能预览图

同时，该功能也支持终止条件设置，可设置为邮箱、电话、安全事件、黑客组织，支持多选。对搜索路径也提供最短路径、有向路径两种配置。如图 5.10 所示是对 IP>175.168.46.204 的一键溯源，通过配置框可知：该节点选择的搜索路径既非最短路径，也不是有向路径，溯源终止条件为邮箱、电话、安全事件、黑客组织四种节点，最终追踪到关联邮箱“devilangel1004@gmail.com”和黑客组织“Hacking Team”、“NIS-5163-Troops”。与图 5.9 相比，可以发现一键溯源功能，可以快速找到源头，获取攻击者的真实社会信息，但其仅能展示关键路径信息，并不能反映 Hacking Team 整个黑客组织庞大复杂的关联关系。对于安全分析人员来讲，通过一层一层地向下关联钻取，往往可以获取一些机器无法分析的信息，发现潜在威胁，这也是关联钻取功能的不可替代性。

5.3.2 力引导-退火算法测试

对交互式关联钻取可视化方法中的力引导-退火算法做节点达到稳定性的速率测试，在页面分别加载 100，200，300，500 个数据节点，观测节点达到稳定性的时间，如下图 5.11 所示。

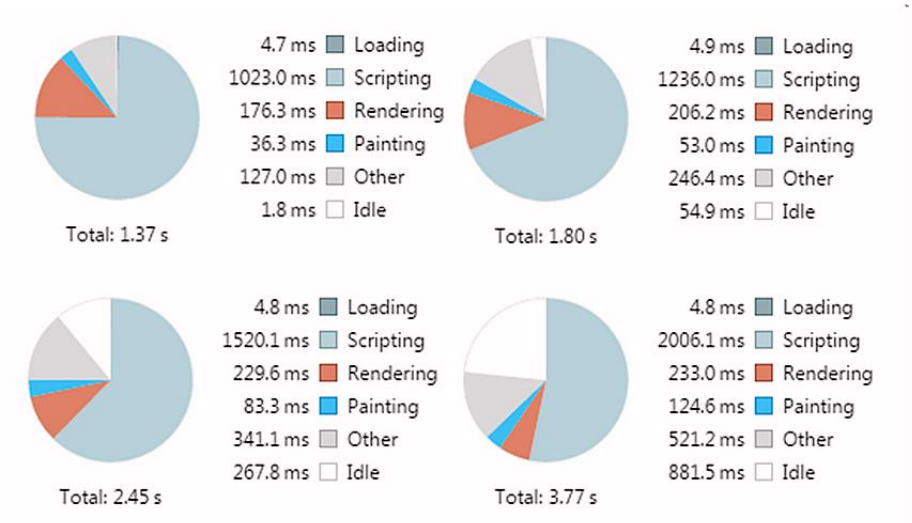

图 5.11　不同数量节点达到稳定性所需时间

通过对威胁情报不同节点个数做关联关系展示，发现图形达到稳定时间符合第三章中力引导-退火算法算法对不同节点数达到图形稳定的数值判定，验证了力引导-退火算法的有效性。

5.4　情报维护模块

5.4.1　情报导入导出功能

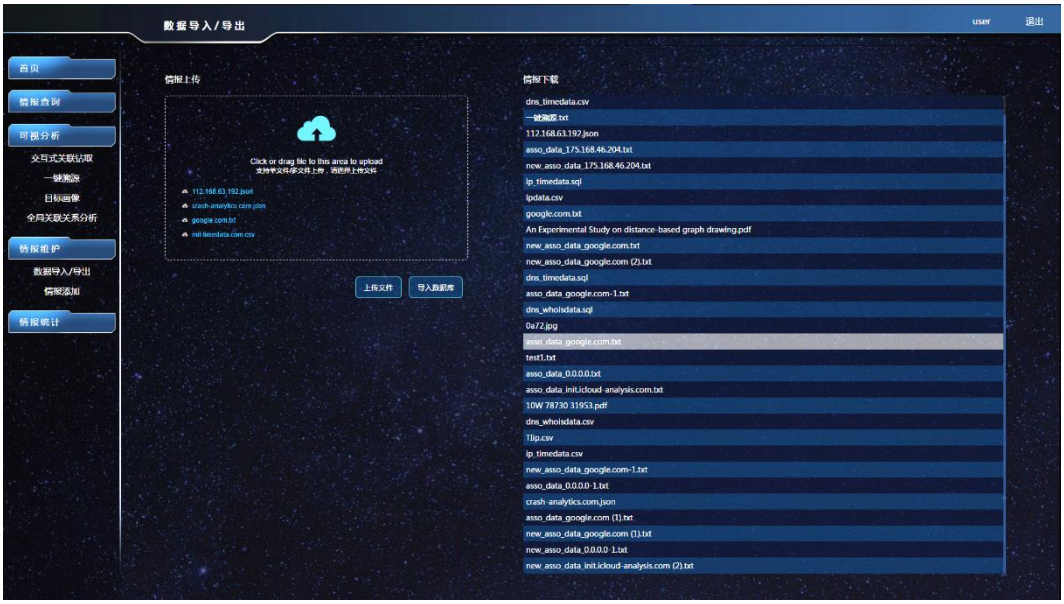

图 5.12　威胁情报可视化系统情报导入导出界面

情报导入导出功能是情报共享的形式之一，用户将获取的安全情报信息以文件的形式上传到服务器，后台管理人员定时对上传的文件进行审核，审核通过的文件以列表的请示在网页展示，当其它用户对某条威胁情报信息有需求时，点击表格中相应的文件名，便可将文件下载到本地。情报的上传下载功能采用组件化的开发模式，情报上传组件的支持单点、多点文件上传，对上传文件进行格式化处理，获取相应的信息，例如文件名，文件种类，文件大小等，用异步传输的方式发送给后台，该组件实现难度不大，主要工作量在于对请求和各类触发事件的封装。情报下载组件以对象数组的形式返回给浏览器文件名列表。用户点击相应的文件名，浏览器向后台发送 get 请求，后台返回文件链接，直接下载到本地。如图 5.12 所示，是威胁情报可视化系统导入导出页面的真实预览图。

5.4.2 情报添加功能

情报添加是另外一种情报共享形式，威胁情报信息日新月异，及时更新威胁情报信息和添加新的情报信息有助于丰富威胁情报基础库。如图 5.13 所示，当用户发现新的情报源时，可填写情报的基本信息，提交给后台。后台对情报来源以及情报可信度进行评估，将来源可靠、可信度高的情报添加到数据库。该页面使用 Element-UI 实现组件的快速开发，可以高效快捷地获取每个输入框的数据，进行整合，以 JSON 格式提交给后台。相比之下，使用传统的 input 标签，需要给每个标签定义 id，并采用 DOM 形式获取每个标签的数据，开发繁琐复杂，复用性极差，不利于迭代开发。

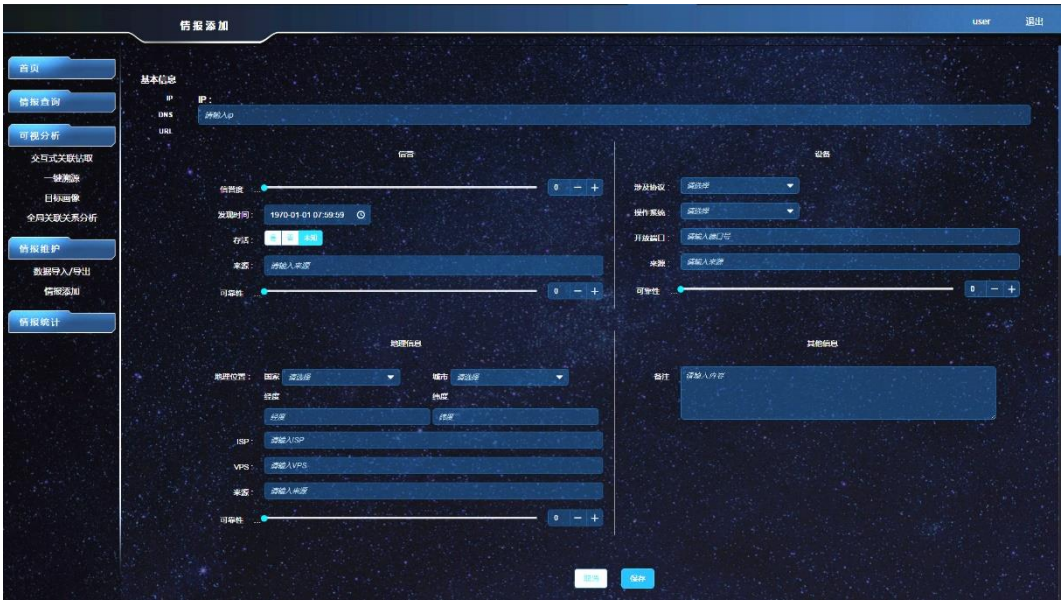

图 5.13 威胁情报可视化系统情报添加功能界面

5.5 情报统计模块

情报统计模块主要是对威胁情报基础库的数据量进行统计，如下图 5.14 所示，威胁情报内链库的数据已达到 1331 万条，其中，IP 库占比 78.06%，DNS 库占比 14.97%，URL 库占比 6.97，MD5 库占比 0.04%。外链库情报已融合一大半知名国内外威胁情报厂商提供的数据。右边是对数据统计进行可视化展示，可以直观的发现外环绿色模块表示外链库已能获取数据的厂商，红色模块表示尚未获取数据的厂商，内环表示内链库数据中 IP、DNS、URL、MD5 的占比。

图 5.14 威胁情报可视化系统情报统计功能

6 总结与展望

6.1 本文工作总结

本论文对威胁情报数据提出了可视化的设计方法及其系统实现，其特点是：

（1）针对性：针对威胁情报数据的多维、多源、异构的特性提出可视化设计方案。

（2）新颖性：提出了交互式关联钻取可视化方法，利用力引导-退火算法优化可视化展示效果，使用 WebSocket 协议实现新增数据动态载入。

（3）高效性：提高关系图布局算法时间效率，优化威胁情报可视化展示策略，能够帮助安全分析人员高效地分析数据以及快速感知威胁。

本文针对性地提出了对多维、多源异构数据适用的可视化方法，设计并实现了威胁情报多源异构数据的交互式关联钻取技术，利用可视化的方式呈现数据中隐含的信息与威胁的发展规律，提升安全分析人员对威胁情报的解读效率，进而辅助决策。

6.2 本文研究工作展望

6.2.1 存在不足

将威胁情报可视化系统进行部署并应用在实际情景中，通过用户反馈意见，总结了以下几个需要改进的方面：

（1）在安全分析人员进行逐层关联钻取的过程中，节点数据不断累积，当数据过多时容易影响用户的分析判断，因此，如何合理地隐藏不必要的信息是一项需要优化的工作。

（2）用户操作回退是本系统尚未实现的一个技术难点。

（3）当安全分析人员进行关联钻取操作时，应对用户行为特点进行采集，通过大数据分析技术生成智能推荐功能。

6.2.3 今后改进的方向

对于数据可视化的未来研究方向，本论文设计的可视化方案以后可以结合其他的技术内容。

（1）威胁情报的可视化可以结合语义分析及本体研究，将可视化与可视分析相结合，数据中隐含的信息将会被一步一步挖掘出来，帮助用户做更好的决策。

（2）在可视化方面，对布局算法进行时间复杂度优化和布局效果优化仍是一项需要继续进行的工作。在用户交互方面，需要不断完善功能，如缩放功能，用户回退功能，智能推荐功能等。

参 考 文 献

[1] Gartner. Definition: Threat Intelligence[EB/OL]. [2015,10,08]. https://www.gartner.com/doc/2487216/definit-eon-threat-intelligence.

[2] 单琳. 网络威胁情报发展现状综述[J]. 保密科学技术,2016,(08):28-33.

[3] Martins R M, Andery G F, Heberle et al. Multidimensional Projections for Visual Analysis of Social Networks[J]. Journal of Computer Science & Technology, 2012(04):791-810.

[4] Best D M, Endert A, and Kidwell D. “7 key challenges for visualization in cyber network defense”[C]. in Proceedings of the Eleventh Workshop on Visualization for Cyber Security. New York, NY, USA: ACM, 2014, pp. 33-40.

[5] Lee D, Song I S, Kim K J et al. “A study on malicious codes pattern analysis using visualization”[C]. 2014 International Conference on Information Science & Applications, 2011, pp. 1-5.

[6] Chen S, Guo C, Yuan X et al. “Oceans: Online collaborative explorative analysis on network security”[C]. in Proceedings of the Eleventh Workshop on Visualization for Cyber Security, New York, NY, USA, 2014, pp. 1-8.

[7] Fischer F and Keim D A. “Nstreamaware: real-time visual analytics for data streams to enhance situational awareness”[C]. in Proceedings of the Eleventh Workshop on Visualization for Cyber Security, Paris, France, November 10, 2014, pp. 65-72.

[8] Chandra J V, Challa N, Pasupuleti S K. Intelligence based defense system to Protect from advanced persistent threat by means of social engineering on social cloud platform[J]. Indian Journal of Science and Technology, 2015, 8(28): 63544.

[9] Carvalho V S, Polidoro M H, Magalhaes J P. “OwlSight: Platform for Real-time Detection and Visualization of Cyber Threats”[C]. IEEE International Conference on Intelligent Data and Security, New York, NY, USA: IEEE, April 2016.

[10]林晨希,薛丽敏,韩松. 浅析网络安全威胁情报的发展与应用[J]. 网络安全技术与应用,2016,(06):12-13+15.

[11]杨泽明,李强,刘俊荣,刘宝旭. 面向攻击溯源的威胁情报共享利用研究[J]. 信息安全研究,2015,(01):31-36.

[12]王慧强,赖积保,胡明明,等. 网络安全态势感知关键实现技术研究[J]. 武汉大学学报：信息科学版, 2008(10):995-998.

[13]王卓君. 一种用于情报威胁评估的数据分类算法研究[J]. 情报杂志,2011,(10):156-162+177.

[14]吕宗平,钟友兵,顾兆军. 基于攻击链和网络流量检测的威胁情报分析研究[J]. 计算机应用研究,2017,(06):1-5.

[15]管磊,胡光俊,王专. 基于大数据的网络安全态势感知技术研究[J]. 信息网络安全,2016,(09):45-50.

[16]李骏韬,施勇,薛质. 基于 DNS 流量和威胁情报的 APT 检测[J]. 信息安全与通信保密,2016,(07):84-88.

[17]李杰. 基于图的实体关联关系可视化技术研究[D].国防科学技术大学,2014.

[18]Ham F, Rogowitz B. Perceptual Organization in User-Generated Graph Layouts[J]. IEEE Transactions on Visualization and Computer Graphics. 2008, 14(6): 1333-1339.

[19]Eades P. A heuristic for graph drawing[C]. In Congressus Numerantium,1984(42):149-160.

[20]Kamada T, Kawai S. An algorithm for drawing general undirected graphs[J]. Processing Letters archive Volume 31,Issue 1 (April 1989) table of contents. Pages：7-15.

[21]Fruchterman T M J, Reingold E M. Graph Drawing by Force-Directed Placement[J]. Software -Practice and Experience 21,11(1991),1129-1164.

[22]Davidson R, Harel D. Drawing Graphs Nicely Using Simulated Annealing[J]. ACM Transactions on Graphics,1996(15),4:301-331.

[23]Frick A, Ludwig A, Mehldau H. A Fast Adaptive Layout Algorithm for Undirected Graphs[J]. Lecture Notes in Computer Science,1995, Volume 894/1995,388-403.

[24]Harel D, Koren Y. A Fast Multi-Scale Method for Drawing Large Graphs[C]. Proceedings of the Working Conference on Advanced Visual Interfaces,2000:282-285

[25]Harel D, Koren Y. Graph drawing by high-dimensional embedding[J]. Journal of Graph Algorithms and Applications, 2004(8)2:195-214.

[26]Shijiang Hou. Information visualization, physicality and intuitive use for tangible user interfaces[J]. Computer Aided Drafting, Design and Manufacturing, 2012(03):18-22.

[27]钟明. 网络安全数据可视化探究[J]. 网络安全技术与应用,2015,(01):118+121.

[28]Li C F. Research on Consistency between Mental Model and Information Visualization and Practice in primary school Sign System Design[A]. 信息化与工程国际学会.Proceedings of 2015 4th International Conference on Computer, Mechatronics, Control and Electronic Engineering(ICCMCEE 2015)[C].信息化与工程国际学会,2015:6.

致　　谢

在论文即将完稿之时，心中感慨万分。在此，我想对在毕设完成过程中给予过我帮助的老师和同学，表示衷心的感谢。

感谢中国科学院信息工程研究所的卢志刚老师、刘俊荣、姜政伟老师，感谢你们的教诲和指导，在论文选题、技术路线确定、论文写作、论文修改等多个环节，先后提出了许多宝贵的意见和建议。正是老师们的教诲和严格把关，让我少走了很多弯路。

感谢李豪杰老师，老师依他渊博的知识、高尚的人格魅力、严谨的治学态度和对学术的奉献精神为我确立了好的写作题材，理清了论文的思路，明确了研究方向。感谢老师在论文定稿过程中，为我提供大量的修改意见，使我顺利完成论文。

感谢课题组师兄师姐的帮助，江钧师兄不厌其烦，一次次地给予我指导，在我遇见工程开发难题时，提供给我新的思路；在我遇见学术瓶颈迷茫不知所措时，指引我前进的方向。感谢课题组的田甜、陈璐、刘松、杜翔宇、陈明毅等师兄师姐，你们在我的学习生活中给予我许多帮助和支持。

感谢我的母校大连理工大学，在这里我走进了编程的大门，学到了许多实用的专业知识；感谢软件学院领导、老师们四年来对我无微不至的关怀和照顾，是你们的诲人不倦造就了现在的我；感谢软 1301 班和 OurEDA 实验室的同学们，是你们在我遇到困难时一如既往地支持我，给我探索、奋进的力量。

感谢我的父母和所有亲人，感谢你们不辞辛劳地将我抚育成人，我爱你们。

最后，感谢正在评阅此论文的老师，您辛苦了。