未来网络发展大会
FUTURE NETWORK
DEVELOPMENT CONFERENCE
NANJING · CHINA / 中国 · 南京

未来网络技术发展系列白皮书（2025）

网络原生智能架构
重构安全网络一体化白皮书

第九届未来网络发展大会组委会

2025年8月

版权声明

编写说明

主要编写单位：

紫金山实验室

主要编写人员：

逯云松、吴柯萌、黄一凡、刘超、郭栋、薛妍妍、李天萁、赵倩、周序、李煊、王晓露、冉茂莹

前　言

随着全球信息通信技术的快速发展，网络的智能化水平不断提升，网络智能化已成为核心研究方向之一，传统的网络架构在面对日益复杂的应用场景和多样化的用户需求时，逐渐显现出其局限性。面对云、边、物多元场景交汇的挑战，我们提出一种全新的架构思维——**网络原生智能（Network-Native Intelligence）**，在安全网络一体化的基础上，可在网络自身之中、之上原生地生长出 AI 能力。

本白皮书正式发布“**网络原生智能架构**”。该架构**基于图建模与推理能力的可编排智能框架**，具备高度解耦、自适应与跨域感知能力。框架天然融合了网络拓扑、安全策略、业务意图等要素，以图为基础、以编排为方法，可实现安全能力的智能组合与按需投送。

在此架构之下，网络与智能不再是串联关系，而是**共生于一体的协同系统**。白皮书详细阐述该架构的设计理念、技术基础、核心能力、应用场景及行业落地价值，并引领行业进入“安全网络一体化”的新范式。

本白皮书介绍了现有网络原生智能的背景与挑战，描述了网络原生智能的设计理念，通过安全网络一体化机制，颠覆传统网络与安全割裂模式，强调二者一体化共生，以智能为核心驱动，安全能力在网络中自然生长。技术基础以图建模和编排方法为核心，融合轻量级 AI 模型、可扩展机制，实现高度解耦与自适应能力。核心能力涵盖感知、

理解、决策和响应四个阶段，支撑安全能力的智能组合与按需投送。应用场景聚焦工业 4.0 时代的安全防护案例，结合云、边、物多元场景，提供针对性防御策略。行业落地价值体现在资源优化、专用性、可控性，引领“安全网络一体化”新范式，应对现代网络攻击的复杂性并提升整体系统防护效能。

目　　录

一、背景与挑战

1.1 研究背景

人工智能（Artificial Intelligence，AI）是利用计算机或者由计算机控制的机器，模拟、延伸和扩展人类智能的理论、方法、技术及应用的一门新技术科学。随着大数据、云计算技术的普及，分布式存储和计算能力的大幅提升，人工智能在多个领域得到了快速应用，例如语音识别与合成、计算机视觉、知识图谱、自然语言处理、人脸识别、机器翻译、舆情分析、推荐系统、自动驾驶等。近年来美国政府在《国家人工智能研发战略规划》的基础上，发布了《国家人工智能研发战略计划》，提出了 8 项国家人工智能研发战略，确定了联邦政府在人工智能研发方面投资的优先领域，以不断提升美国的人工智能应用能力[1]。其他国家也相继将人工智能技术提升到国家科技发展的战略高度，人工智能必将越来越深入地渗透到各行各业和社会生活的方方面面。它涉及的范畴包括自然语言处理、智能搜索、推理、规划、机器学习、知识获取、模式识别、神经网络、遗传算法等。人工智能的核心是算法，包括传统的机器学习算法和非传统的机器学习算法，其中，传统的机器学习算法主要解决简单的应用场景以及结构化的数据，非传统的机器学习算法主要解决比较复杂的应用场景以及非结构化的数据或者多样化的数据。

全球正步入一个以数字化、网络化、智能化为核心特征的全新发展阶段[2]。以“新型基础设施建设”（简称“新基建”）为代表的宏伟蓝图[3]，正在全球范围内重塑经济社会的发展根基。从高速泛在的5G 网络、工业互联网，到支撑海量数据处理的人工智能与云计算中心，“新基建”不仅是技术设施的迭代升级，更是一场深刻的结构性变革，其最终目标是构建一个万物互联、数据驱动、智能引领的社会运行体系。以中国的“东数西算”国家工程为例，其构建了一个横跨东西、服务全国的一体化算力网络，旨在优化国家算力资源布局，为千行百业的数字化转型提供澎湃动力[4]。

在应用人工智能技术的各个行业中，网络安全是活跃度排名前 3 的行业之一，典型应用例如恶意流量识别、钓鱼邮件检测、恶意代码识别、僵尸网络检测等[5]。近年来，网络空间安全重大事件持续爆发，网络安全威胁全面泛化。斯诺登事件、乌克兰电网攻击事件、美国大选干预事件等表明，网络空间安全威胁覆盖了从物理基础设施、网络信息系统到社交媒体信息，对虚拟世界、物理世界的诸多方面构成威胁。网络空间安全已经成为非传统安全的重要组成部分。随着人工智能第三次浪潮的兴起，人工智能向诸多行业、领域不断渗透并交叉融合的趋势已经显现。人工智能因其智能化与自动化的识别及处理能力、强大的数据分析能力、可与网络空间安全技术及应用进行深度协同的特性，对网络空间安全的理论、技术、方法、应用产生重要影响，促进变革性进步。着眼人工智能赋能网络攻击的威胁和影响，从防范安全威胁、构建对等能力的视角着手，尽快开展重大关键技术研究。推

动“产学研”机构以有效应对人工智能赋能攻击新型威胁场景为首要需求，从攻防两方面进行联合攻关，开展智能化威胁态势感知、自动化漏洞挖掘与利用、智能恶意代码等技术研究。加快人工智能技术在国家、重要行业关键信息基础设施安全防护方面的体系化应用，整体性完成智能化升级换代，大幅提升关键信息基础设施安全保障、网络安全态势感知、网络安全防御、网络威慑的能力水平。

1.2 网络原生智能现状及挑战

随着全球信息通信技术的快速发展，网络的智能化水平不断提升，特别是在即将到来的 6G 时代，网络智能化已成为核心研究方向之一[6]。传统的网络架构在面对日益复杂的应用场景和多样化的用户需求时，逐渐显现出其局限性。为了解决这些挑战，学术界和工业界开始探索“网络原生智能”（Network Native Intelligence，NNI）的概念，即将智能转变为一种可按需生成、精准交付的网络原生能力，以满足未来应用对高阶智能服务（如分布式 AI、安全网络一体化）的根本性需求[7]。

在 5G 网络中，虽然 AI 技术已经开始应用于部分网络功能，如网络负载预测和用户行为分析，但其整体架构仍主要依赖于传统的网络功能分离（Network Function Virtualization，NFV）和软件定义网络（Software Defined Networking，SDN）等技术。相比之下，6G 网络则将更深入地融合 AI 技术，致力于构建一个“AI 原生”（AI-Native）的网络智能系统，从而实现更高层次的智能化管理[8][9]。这种融合不

仅体现在网络核心功能的智能化改造上，还涉及网络架构的整体重构，以及跨域协作机制的深入研究。

网络原生智能并非简单地引入 AI 模型以优化网络基础设施，而是代表了对网络角色的一次范式重塑：即将网络从被动的数据流水线，转变为一个能够主动为分布式智能任务提供原生执行环境的计算平台。在这一前瞻理念下，如 DAEMON 项目所展示的，其技术关键在于构建一个网络智能协调器（Network Intelligence Orchestrator）[10]。该协调器并非传统意义上的网络控制器，而是作为智能工作负载的生命周期编排引擎，它通过对底层网络资源的深度抽象与统一调度，实现了 AI/ML 模型部署、执行与协同的自动化。其所展现出的灵活性与适应性，本质上是网络按需组合与交付智能服务的核心能力体现。

然而，尽管网络原生智能具有广阔的应用前景，其在实际部署过程中仍面临诸多挑战。首先，AI/ML 模型在网络中的应用需要解决模型依赖性和跨域协作等复杂问题。尤其是随着 6G 网络的引入，网络功能间的模型依赖性将更加复杂，如何在保障网络服务质量的同时，合理管理这些依赖性成为一个亟待解决的问题[11]。此外，6G 网络中的智能化管理还需要进一步优化跨域协作机制，以实现各个智能节点之间的高效协同。

因此，需进一步将网原智能工作前置，探索用多模态在线训练与网络智能模拟等前沿技术，构建“网络原生智能”系统，构建源于网络，服务于多形态、多业务 AI 的“网络原生智能”，挑战网络智能，建立面向 AI 服务与 AI 应用的端到端安全防御系统，保障 AI 数据安全、

模型安全、服务安全以及应用安全。

二、网络原生智能理念

2.1 网络原生智能的定义

网络原生智能是一个以图建模为核心，通过智能编排引擎，调度原生于网络设备中部署的 AI 能力，并利用实时反馈进行持续优化的自动化协同框架。其主要理念是在安全网络一体化的基础上，可在网络自身之中、之上原生地生长出 AI 能力。该架构是一个基于图建模与推理能力的可编排智能框架，通过深度融合网络拓扑、安全策略与业务意图，最终实现安全能力的智能组合与按需投送。具体架构图

图 1-1 网络原生智能架构图

如图 1-1 所示。

2.2 网络原生智能的核心特征

网络原生智能的核心特征如下：

图 1-2 网络原生智能的核心特征

（1）泛在的分布式智能

网络原生智能框架主张 AI 工作负载（包括模型训练和推理）应根据成本效益分析，被部署在网络中最合理的位置，无论是终端设备、网络边缘，还是中心云。这种“智能无处不在”的理念，打破了传统集中式 AI 模型的束缚，是实现低延迟应用和高效资源利用的关键。网络本身演变为一个巨大的、分布式的计算平台，智能分析能力可以根据任务需求，被灵活地调度到离数据源最近的地方。

分布式智能技术主要分为分布式智能训练与分布式智能推理两方面，在分布式智能训练领域，联邦学习是一种经典架构。参与训练的客户端无需上传本地数据，仅需上传训练后的 ONNX 模型更新；边缘服务器节点对这些模型参数聚合更新后，再下发给各客户端。分布

式智能推理指在网络边缘分布式执行 ONNX 模型。由于边缘节点的计算与存储资源有限，如何减小并优化模型在分布式推理中的开销显得尤为重要。常见的模型压缩方法包括网络剪枝、知识蒸馏、参数量化、结构优化等[12]。

（2）持续学习与实时自适应

网络原生智能框架能够在无需人工干预的情况下，实时地学习和适应网络环境变化。这通过架构中内建的闭环反馈机制得以实现。这些闭环持续捕捉网络交互和运营结果，自动将这些反馈用于模型的迭代和优化，使系统能够“自我进化”。面对 6G 网络环境的高度动态性和不可预测性，这种持续学习和自适应的能力是维持网络高性能和高可靠性的根本保障。

（3）动态任务图调度

网络原生智能框架通过分布式任务图调度技术，将复杂的安全任务细分并分配到不同的任务图中进行处理，这不仅优化了计算资源的利用效率，还显著提高了系统的吞吐能力和响应速度。其中每个任务图都可以独立运行，并根据不同的需求进行动态调整。这种设计灵活性使得系统能够根据网络环境的变化进行即时的优化配置，从而显著提升系统的适应能力和防护效果。例如，当检测到新的安全威胁时，系统可以即时加载新的防护策略或调整现有任务的执行顺序，而无需重新部署整个系统。这种即时反应能力在当今瞬息万变的网络环境中尤为重要，它确保了任务图驱动能够始终保持在最佳防护状态。

4）多阶段并行流水线

在网络原生智能框架中，多集安全防御策略的引入是实现高效网络安全防护的重要组成部分。通过多阶段并行流水线设计，将网络流量从初步分析到对流量进行处理的过程，分为感知、理解、决策和响应。在多集安全防御策略基础上，网络原生智能框架还通过引入网络业务的服务水平协议（Service Level Agreement，SLA）优化模型，实现了在算力网络中的 SLA 协议优化。

2.2 网络原生智能的安全基础

网络原生智能包含两大安全基础，分别为智能驱动安全和网络安全一体化：

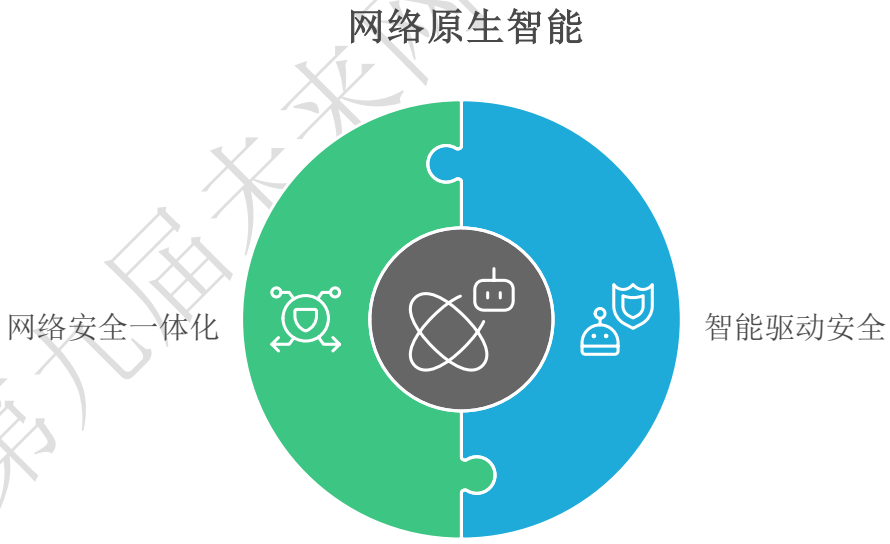

图 1-3 网络原生智能技术组成

（1）智能驱动安全

传统基于机器学习的安全检测安全技术通常依赖于预设规则和人

工分析，面对日益复杂且快速演变的威胁时，响应滞后、误报率高且难以应对未知攻击，形成“事后诸葛亮”式的被动防御。通过遍布全网的分布式智能体协同工作，网络能够自主地感知安全威胁、深入分析潜在风险并迅速作出决策，从而实现从传统的“亡羊补牢”式被动防御向“未雨绸缪”式主动预测与防护的根本性转变，构建起一个具备全局视野和快速响应能力的智能安全防护系统。

（2）网络安全一体化

传统网络安全领域，安全产品和网络往往各自为政，形成分散且孤立的“烟囱式”架构，导致安全信息难以共享、策略难以协同，安全事件响应效率低下，甚至出现安全盲区。网络安全一体化技术旨在打破这些壁垒，通过构建一个统一协同的安全防护系统，实现安全能力的内聚与联动。该系统不仅能实现安全事件的自动化感知、深度分析和智能处置，更强调将处理结果实时反馈至整个网络，从而形成一个自适应的安全防御能力，最终构建出能够全面抵御复杂威胁的“免疫系统”。

2.4 网络原生智能的概念对比

2.4.1 Network for AI 和 AI for Network 的对比

Network for AI 和 AI for Network 代表了人工智能与网络基础设施融合的双向路径，前者强调网络架构的优化以支持人工智能应用的运行，而后者则聚焦于利用人工智能技术来提升网络本身的性能和管

理效率，这种区别不仅体现了技术发展的互补性，还突显了从基础设施到应用优化的全面转型[15]。Network for AI 主要关注设计和构建网络系统来满足人工智能工作负载的需求，例如通过高带宽、低延迟的互联技术如 InfiniBand 或优化以太网来处理大规模数据传输和计算任务[16]，这使得它特别适用于数据中心和边缘计算环境，其中 GPU 集群需要高效的分布式计算支持，以实现人工智能模型的训练和推理过程，从而提高整体系统效率并减少瓶颈[17]。相比之下，AI for Network 则将人工智能算法作为工具嵌入网络管理中，例如采用 AIOps（人工智能运维）来实现故障预测、资源动态分配和自动化配置[18]，这有助于网络从传统的反应式维护转向预测式优化，显著降低停机时间并提升安全性。

总体上，这种区别推动了行业从单一方向的优化向闭环生态的构建演进[19]，在实际应用中，二者往往相互依赖，例如在 6G 网络中，Network for AI 提供支持人工智能的底层架构，而 AI for Network 则利用这些架构进行自我优化，从而形成一个高效、适应性的智能网络体系。

2.4.2 网络原生智能与 AI for Network 的关系

在人工智能与网络基础设施深度融合的背景下，网络原生智能与 AI for Network 之间形成了紧密的驱动与优化关系，前者强调人工智能算法直接嵌入网络架构中成为其内在组成部分，实现分布式智能代理的协作和实时适应，而后者则聚焦于利用人工智能技术来提升网络

的管理效率、性能和自动化水平，这种关系不仅体现了从外部工具到内在嵌入的演进路径，还为网络从被动传输向主动智能决策的转型提供了关键机制。

网络原生智能的核心在于将 AI 作为网络的“原生”功能，例如通过边缘计算和分布式学习机制在无线接入网或核心网中部署模型[21]，从而推动网络的韧性和规模化协作，而 AI for Network 则为其提供必要的优化工具，如 AIOps 和机器学习算法，用于故障预测和资源动态分配，确保嵌入式 AI 模型能在复杂环境中高效运行[22]。进一步而言，这种关系在 5G-Advanced 和 6G 愿景中尤为突出，AI for Network 通过预测式维护和自动化配置机制，使网络原生智能能够实现“零接触优化”的目标，例如支持联邦学习以保护隐私并加速模型迭代，从而提升网络的自主性和可靠性[23]。总体上，AI for Network 不仅是网络原生智能的实现路径，还促进了其在实际应用中的扩展，例如在电信运营商环境中，AI 驱动的流量分析确保嵌入式智能的无缝协作，而网络原生智能则利用这些工具进行实时决策，形成一个闭环的生态循环。这种关系还延伸到标准制定中，例如在 3GPP 规范中，强调 AI for Network 的算法框架作为网络原生智能嵌入的支撑，推动从传统网络向 AI-centric 架构的平滑过渡[24]。在具体实践如 Nokia 的 AI-native 框架中，这种关系表现为双向强化[28]，AI for Network 不仅提供监控和根因分析以支持网络原生智能的部署，还通过 MLOps 机制确保模型的生命周期管理，进一步降低了复杂性和数据隐私挑战。

2.4.3 网络原生智能和智能原生网络的对比

“智能原生网络”与“网络原生智能”则共同代表了人工智能与网络技术深度融合的前沿方向，然而，两者在架构设计与研究对象上存在本质区别。智能原生网络是专为满足大规模 AI 工作负载通信需求而设计的网络架构，遵循 Network for AI 的设计原则。其主要目标是通过优化数据传输来提升 AI 集群的整体计算效率，这一效率通常通过 AI 任务完成时间和 GPU 利用率等指标进行评估[26]。为实现此目标，该网络架构致力于构建一个为 AI 服务的可预测的、端到端的无损以太网[27]，其关键技术包括：采用支持远程直接内存访问（Remote Direct Memory Access over Converged Ethernet，RoCE）v2 的硬件，以及针对 AI 训练中常见的集合通信流量模式而优化的拥塞控制和负载均衡算法[28]。在实际部署中，智能原生网络的设计涵盖了从交换机、DPU 智能网卡到网络操作系统和管理软件的整个技术栈，旨在与 AI 计算单元紧密配合，为连接大规模分布式计算节点的 AI 系统提供必要的、可扩展的高吞吐量与低延迟通信能力。

而网络原生智能致力于提升网络自身的智能化水平（AI for Network）。如果说“智能原生网络”的目标是为 AI 应用构建一条极致通畅的网络，那么网络原生智能的目标则是让网络本身具备思考、感知和协同行动的能力。它不以加速 AI 训练任务或提升 GPU 利用率为主要目的，而是追求网络在安全防护、自动化运维和资源调度等原生能力上的革新。

在实现路径上，二者的技术栈存在显著差异。智能原生网络聚焦于物理层和传输层的技术，如通过采用 RoCEv2 和专门的拥塞控制算法来打造无损、低延迟的数据通道[29]。相比之下，网络原生智能更侧重于架构和软件层面的创新。它通过引入图建模技术来统一描述网络拓扑、安全策略与业务意图，形成一个可供机器理解和推理的任务流水线。基于此，它将 AI 能力（如安全检测模型、流量分析模型）作为可调度的插件功能，通过一个智能编排引擎，动态地部署到网络中的路由器、交换机或边缘节点上，使整个网络成为一个分布式的 AI 计算平台。

网络原生智能与相关概念的对比如表 2-1 所示：

方面	网络原生智能	智能原生网络	AI for Network	Network for AI
定义	AI 能力原生于网络，实现分布式主动决策。	为 AI 工作负载通信而设计的专用高性能网络。	利用 AI 技术优化网络运维、性能与自动化。	构建专用网络基础设施以支持 AI 工作负载。
核心目标	网络具备主动决策能力，提升原生安全与韧性。	最小化 AI 任务耗时，最大化 GPU 利用率。	实现网络预测性维护与自动化运维(AIOps)。	为 AI 集群提供无阻塞、低延迟的高速通道。
技术实现	图建模、分布式 AI 调度、边缘模型部署。	无损以太网、RoCEv2、专用拥塞控制。	AIOps 平台、机器学习、强化学习。	高带宽互联技术(如 InfiniBand)。
应用场景	实时安全防御、智能路由、自动化编排。	AI 训练/推理集群、高性能计算(HPC)。	故障预测、自动化运维、资源动态分配。	支撑大模型训练的底层网络。
优势	决策实时性强、韧性高、安全网络一体化。	提升 AI 集群效率、加速模型训练。	提升运维自动化水平、降低故障率。	消除 AI 数据传输瓶颈。
挑战	分布式 AI 的复杂	成本高昂、需与计	模型的准确性与可	大规模集群的拥塞

	性和安全性问题	算深度协同。	解释性。	管理。
关系	AI for Network 的演进与高级实践。	Network for AI 的具体实现。	网络原生智能 的技术使能与工具。	智能原生网络 的设计原则与基础。

表 2-1 网络原生智能概念对比

三、安全网络一体化机制

网络原生智能的目标，是构建能自主感知、决策和执行的网络智能系统。它要求网元设备不仅能智能优化路由、分配资源，更能主动预测风险、自动响应威胁。要实现这一目标，其核心前提是智能系统必须能够获取全面、实时的网络状态信息，并能对网络实施统一、协同的控制，并能预防潜在的安全性问题。然而，如果网络设备与安全设备在物理上分离、功能上独立，则无法在传输中保障模型、数据、业务的安全，这构成了实现网络原生智能的最大障碍。这种分离的架构导致安全策略部署滞后、网络优化与安全需求之间存在冲突，以及资源利用效率低下等突出问题。我们提出“安全网络一体化”这一创新机制，它作为网络原生智能的底层安全支撑，主要通过路由与策略协同以及安全能力按需投送两大核心途径来实现。

3.1 路由与策略协同

3.1.1 传统路由安全面临的挑战

在传统网络中，路由功能与安全策略通常是分离式管理。路由协

议（如 BGP、OSPF）的核心目标是保障网络的连通性与转发效率，而安全策略（如防火墙规则、访问控制列表 ACL）则由独立的网络设备进行配置和执行。

这种分离式架构导致了诸多问题：首先，网络路由的调整是常态，但安全策略的变更却往往无法同步。为了优化路径而更改路由时，安全策略可能因未能及时更新而产生安全盲区，或因不匹配新路径而导致业务中断。在拥有成千上万条规则的大型网络中，人工管理这种动态一致性几乎是不可能的，极易引发策略冲突与配置错误。而且，传统的威胁检测，如旁路部署的入侵检测系统，其工作模式存在固有的延迟。它需要先由网络设备将流量镜像一份，再传输至分析设备，经过分析识别出威胁后，才能产生告警。整个过程链条漫长，从攻击发生到安全团队介入处置，往往存在数分钟甚至数小时的延迟。

更严重的是，这种架构忽视了对网络控制平面自身的安全防护，使得 BGP 路由劫持、泄漏等威胁能够绕过传统安全设备，直接影响网络核心的稳定性与数据流向。

3.1.2 安全网络一体化平台下的路由与策略协同

安全网络一体化平台是指将传统上分离的网络转发、安全防护与运行管控等功能，通过平台进行深度融合的设计范式。其内涵在于：网络基础设施（如路由器、交换机）具备标准化的可编程接口，使其从静态的转发设备转变为可由上层软件定义的策略执行点，使平台能够对全网的设备、策略和流量进行统一的分析、调度与编排。

通过平台，路由与策略协同将安全分析模块的决策结果，转化为网络路由系统可执行的流量调度与路径调整动作的自动化机制。安全网络一体化平台接收来自威胁检测或业务策略模块的高级指令，并将其编译为具体的路由协议操作，从网络层面改变流量的走向，实现对网络行为的动态、精准干预。

以 BGP 路由劫持这一典型的控制平面攻击为例，传统网络对此类威胁的响应严重依赖人工。网络运维团队通常在业务中断或用户投诉后才被动感知，需要通过复杂的 BGP 数据分析来定位劫持源，然后手动登录多台设备配置过滤器进行补救，整个过程耗时数小时甚至数天，期间造成的业务损失已难以挽回。

平台则提供了一种主动、闭环的自动化处置方案。平台的智能感知能力是协同的基础，它通过 BMP 等技术实时采集全网 BGP 路由更新，并与 RPKI 等可信基准进行交叉验证。一旦检测到路由宣告的源 AS 与基准不符，系统便在数秒内判定为“路由劫持”。

此时，平台的协同处置能力将被激活：它不再仅仅是产生告警，而是自动触发处置决策。基于对劫持事件的分析，平台会决策并生成相应的缓解策略，并通过标准化的南向接口下发至网络中的核心及边界路由器。处置方式有如下两种：

- 精确阻断: 平台可生成 BGP FlowSpec 规则，指令全网路由器精准识别并丢弃所有发往被劫持前缀的、且源于恶意 AS 路径的流量。
- 路径重定向: 平台亦可生成一条新的 SRv6 路由策略，将所有访问被劫持前缀的合法流量，强制牵引至一条预设的、可信的、未受

污染的备用路径上，从而在攻击持续期间保障核心业务的连续性。

通过上述协同机制，网络的安全能力不再仅仅是数据平面的被动过滤，而是升级为深入控制平面的、主动的路由路径调度与治理。这种方式打破了传统网络与安全的壁垒，能够更快速、更灵活地应对包括控制平面和数据平面在内的各类安全威胁，在阻断攻击的同时，最大限度地保障正常业务的连续性。

3.2 安全能力按需投送

3.2.1 传统安全能力部署的痛点

传统安全能力在长期实践中暴露了三个主要局限性：

（1）部署僵化与资源利用率低

安全设备的处理能力一旦部署后便难以更改。企业为了应对业务流量的峰值，必须提前采购并部署超出日常需求的硬件容量，导致在大部分时间里，这些昂贵的安全资源处于闲置或低负载状态，造成投资浪费。当业务增长需要扩容时，又面临着复杂的硬件替换、网络拓扑变更和较长的交付周期。

（2）流量路径迂回与性能瓶颈

由于设备集中部署，许多网络流量无法通过最优路径直接到达目的地，而是必须先被重定向至这些安全设备集群，处理完毕后再转发至最终目的地。这种迂回的流量路径显著增加了数据传输的延迟，并占用了额外的网络带宽。同时，这些集中的安全设备的处理能力上限，

也构成了整个网络吞吐性能的瓶颈。

（3）防御能力响应迟缓

传统安全架构下，新的威胁特征库更新、安全策略调整往往需要在每一台独立的安全设备上进行手动配置或分批推送。当面对突发的新型网络攻击（如零日漏洞利用、新型勒索病毒变种等）时，这种分散化的操作模式会导致防御规则无法快速、统一地覆盖整个网络防护节点。此外，对于跨地域、跨网络的分布式业务场景，安全团队需要耗费大量时间协调不同节点的设备参数同步，使得整体防御体系对威胁的响应速度滞后于攻击扩散速度，大幅增加了安全事件的处置难度和潜在损失。

3.2.2 安全能力按需投送理念

“安全能力按需投送”是为解决上述问题而提出的理念，所谓“按需投送”，其本质是依据策略，在数据流经网络设备时，对这些默认处于静默状态的安全功能进行动态“激活”和应用。具体实现上，统一的管控平台负责制定并下发安全策略。当网络设备接收到数据流时，能够实时识别其业务属性或安全风险等级。若该流量匹配了特定策略，设备便会立即调用其内部相应的安全处理模块，在不中断转发流程的前提下完成深度检查或过滤。对于不匹配策略的常规流量，则直接通过高速转发路径处理，其安全模块不被激活，从而避免了性能损耗。这种模式确保了安全防护能够精准、高效地应用于任意节点的任意流量，实现了安全覆盖的无处不在和网络资源的最优化利用，从

根本上改变了传统安全部署的被动和僵化局面。

3.2.3 技术实现

实现按需投送的第一步，是建立一个能够统一制定和下发策略的管控平台。安全网络一体化平台负责将业务或安全需求转化为网络设备可以理解和执行的具体指令。在技术实现上，这依赖于标准化的建模与通信协议。平台采用 YANG 数据模型来对网络设备的安全功能（如访问控制、状态化防火墙、流量过滤等）进行标准化的、结构化的定义。随后，平台通过 NETCONF 协议，与网络设备建立安全、可靠的连接。当管理员在平台上定义一项策略时（例如，“禁止 A 业务群组访问 B 数据库”），平台会将其翻译成符合 YANG 模型的配置数据，并通过 NETCONF 协议以事务化的方式，精准地推送给全网中所有相关的网络设备。这种方式确保了策略能够被准确、一致地部署，并避免了传统命令行配置的复杂性和不确定性。

当策略成功下发至网络设备后，设备必须具备精准识别相应数据流的能力，这是触发“按需”动作的前提。在技术上，这要求网络设备在其入口接口处具备一个高性能的流量分类引擎。传统的五元组（源/目的 IP、源/目的端口、协议号）是基础的分类依据。但为实现更精细化的管控，现代网络设备还需支持更深度的识别技术。例如，通过异常流量检测技术识别特定的流量异常模式，或者根据报文中携带的特定元数据标签（如 VLAN Tag、MPLS Label 或 SRv6 SID 中包含的应用信息）进行分类。当一个数据包进入设备时，分类引擎会

高速匹配这些预设的规则。一旦命中，该数据流便被“标记”并准备接受下一步的策略处理

动态调用是实现“按需投送”的核心环节。当一个数据流被分类引擎成功识别并标记后，设备的控制平面会根据策略指令，动态调用其芯片或操作系统中对应的按需投送功能模块。这个调用过程发生在设备内部，而非将流量转发至外部。例如，若策略要求对一个新建的TCP 连接进行状态化防火墙检测，设备的处理器会为该连接在专门的硬件会话表中创建一个条目，后续属于该连接的数据包将依据此会话表状态进行快速匹配与处理。若策略要求对流向某个服务器的流量进行异常流量清洗，设备则会激活其网络处理单元（NPU）中专门的DDoS 攻击缓解逻辑，对该特定流量进行速率限制和特征过滤。这个“调用”过程是瞬时的，且只针对被标记的流量，确保了常规流量的转发性能不受影响。

3.3.4 按需投送的优势与价值

首先，在网络性能与资源效率方面，该模式旨在减少不必要的性能开销。通过在流量路径上的网络设备进行原生处理，可避免将流量重定向至集中部署的专用安全设备，从而有助于降低因路径迂回产生的网络延迟和带宽消耗。同时，安全功能按需激活的机制，使得设备的安全处理模块在未触发策略时保持较低负载，这种设计旨在提升硬件资源的整体利用效率，并更好地平衡安全处理与高性能转发之间的关系。

其次，在业务响应与部署灵活性方面，此模式提供了一种更为敏捷的能力部署方式。由于安全能力的启用是通过下发软件策略来完成，而非部署实体硬件，因此能够缩短为新业务提供安全防护所需的准备周期，以适应快速迭代的业务环境。它也允许安全策略以更精细的粒度进行应用，例如针对特定的应用或业务流进行差异化配置，这相较于传统的边界防护模型，提供了更为灵活的管控选项。

最后，在安全覆盖的广度和策略的一致性上，该模式也带来了显著改进。由于网络中的众多设备均可作为策略执行点，这种架构具备了将安全防护能力延伸至网络内部的潜力，为传统模型中通常缺乏有效监控的“东西向”流量提供了防护手段。此外，通过统一平台对策略进行集中管理和下发，有助于确保安全规则在不同网络节点间的应用一致性，能够在一定程度上降低因手动、分散配置所引入的策略冲突或遗漏风险。

四、图驱动智能编排的框架设计

在网络原生智能架构中，图驱动智能编排框架扮演着核心中枢的角色，它将网络资源、安全功能和业务需求抽象为图结构模型，通过图推理算法实现要素间的动态关联解析和逻辑决策，从而桥接基础设施与智能应用，确保整个架构从被动响应向主动适应演进。这种关系不仅体现了图驱动机制作为数据基座的支撑作用，还突显了编排引擎在执行层面的关键性。我们以 DDoS 攻击检测和缓解为案例，来阐述

图驱动智能编排框架所完成的过程：首先安全网络数据智能平台先通过可编程交换机的采集技术感知到攻击特征、流量信息、设备状态及环境关联内容，接着由智能分析引擎解析出攻击属性、设备适配性并排除不合规方案，再借助图驱动框架选定“交换机 A 阻断”的方案，确定执行顺序与资源分配，最后通过智能分析引擎下发指令激活设备功能，监控执行状态与效果并反馈更新形成闭环，高效处置了攻击，从而保障了核心业务的正常运行。简言之，通过图驱动与智能编排的框架，使网络设备的安全能力变为可分析、可编排、可升级的标准化模块，成为安全网络一体化中“能力聚合、策略适配、资源协同、快速响应”的智能中枢。

为了构建并实现上述案例中的智能系统，我们需要构建一个全新的、具备高度解耦、自适应和跨域感知能力的框架。本章将深入探讨图驱动与智能编排框架的感知、理解、决策、响应核心能力，以及其在全网流量实时感知与处理能力，拓扑、流量与安全状态的统一图建模能力，可编排智能引擎与动态逻辑能力，解耦 AI 组件与网络设施的插件化机制等关键能力。

4.1 感知、理解、决策、响应的核心能力

图驱动智能编排框架的核心能力由感知、理解、决策和响应这四个步骤构成。具体过程如图 4-1 所示。

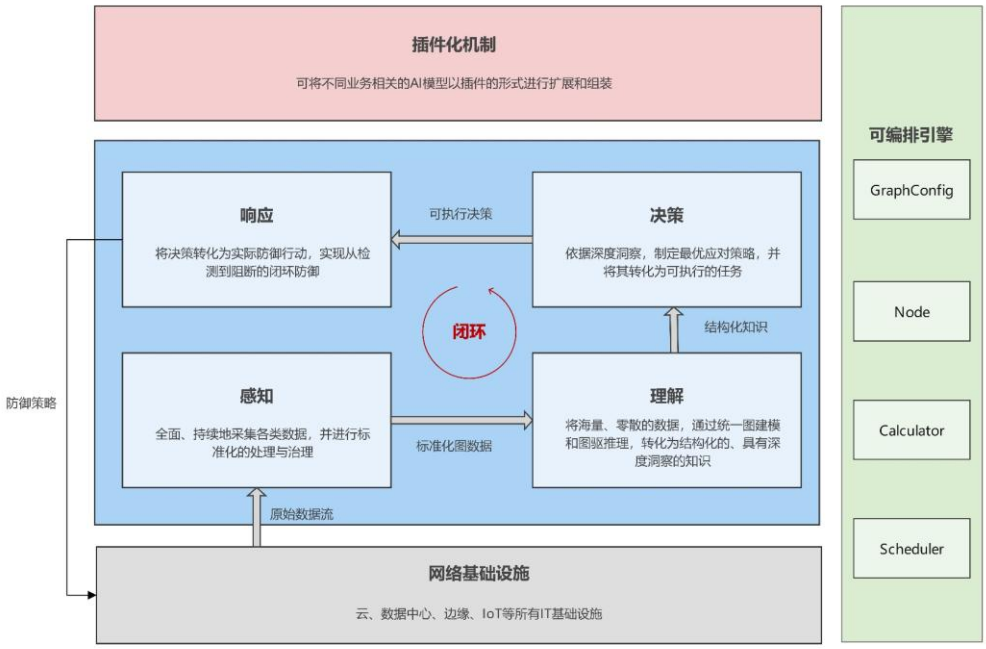

图 4-1 图驱动智能编排的框架图

4.1.1 感知阶段

感知阶段是整个框架的数据基础，其核心职责是从所有相关的网络基础设施中全面、持续地采集各类数据，并进行标准化的处理与治理，为后续的智能分析提供高质量的数据输入。该过程首先通过分布式数据采集能力，从传统数据中心、动态云环境、边缘设备和物联网等多样化的环境中，实时获取全网流量数据。随后，这些原始数据会立即进入实时数据处理与治理流程，进行统一的清洗、格式转换和关

联丰富，特别是对时序数据的处理，以确保数据的一致性与可用性。同时，该层通过持续安全内容监控能力，动态监测用户与设备实体的行为，并对已知的威胁和漏洞信息进行感知，从而确保了数据来源的全面性和安全相关性。

4.1.2 理解阶段

理解阶段的核心任务是将来自感知阶段的海量、零散的数据，转化为结构化的、具有深度洞察的知识。该层通过构建统一图建模来完成这一目标，即将网络拓扑、流量信息、安全实体等关键要素抽象为图的节点，并将它们之间的连接、策略应用和威胁关系等定义为图的边，从而将复杂的网络环境映射为一个统一、关联的数学模型。在此基础上，图驱推理引擎会运用智能编排算法对该图进行深度挖掘，其能力包括执行行为分析以发现偏离正常模式的异常活动，通过攻击路径可视化直观地展现潜在攻击的传播路径，以及进行威胁情报融合，将外部威胁数据与内部网络状态相结合，最终形成对安全态势全面而深刻的理解。

4.1.3 决策阶段

决策阶段依据理解阶段提供的深度洞察，负责制定最优的应对策略，并将其转化为可执行的任务。首先，智能决策引擎基于跨域情报分析和 AI 辅助决策技术，对识别出的风险进行风险定级，并能够根据预设的业务或安全意图，自动生成相应的安全策略，即意图驱动策

略生成。决策制定后，可编排智能引擎会负责后续的执行与协调。它通过策略解析功能，将抽象的策略指令翻译成具体设备能够识别的命令，并可调用预设的自动化剧本来执行标准化的操作流程，最终通过任务下发机制，将指令准确无误地传递给响应阶段。

4.1.4 响应阶段

响应阶段是将决策阶段生成的策略转化为实际行动的关键环节，核心目标是依托网络设备的策略执行能力，实现对威胁的快速阻断、流量的动态调控及安全状态的持续优化，最终形成从检测到阻断的闭环防御。在安全网络一体化的解决方案中，通过路由器为核心执行单元，融合硬件加速、标准化接口与状态反馈机制，形成从策略下发到效果验证的完整闭环，确保安全意图在网络中高效落地。其能力包括如下 3 个方面：

（1）路径与应用的精细化控制

为实现有效的策略协同，执行单元需具备精准的控制能力。SRv6 技术通过网络路径可编程性，能够将抽象的策略意图（如租户隔离）转化为具体的数据转发行为，确保不同业务的流量严格按照预设路径进行端到端传输。例如，在应对路由劫持时，可通过下发 SRv6 策略，将受影响的业务流量强制牵引至一条可信的备用路径，保障业务连续性。

（2）数据平面的策略执行与状态反馈

路由器的 ACL 能力是策略执行的直接体现。它支持在硬件层面对

数据流进行过滤和阻断。同时，现代路由器能够为 ACL 规则关联独立的硬件丢包计数器。当有报文因匹配 Deny 规则被丢弃时，相应的计数器会自动累加。上层管控平台可通过周期性地查询这些计数器，精确地量化策略的执行效果，为实现自动化闭环处置和安全态势分析提供了数据输入。此外，更先进的 BGP FlowSpec 技术，允许平台基于 BGP 属性动态生成流量过滤器，为处置路由安全威胁提供了更精准的手段。

（3）自动化管控的标准化接口

上述能力的调度协同，依赖于统一、开放的管控接口。以 NETCONF 协议及相应的 YANG 数据模型为代表的标准化南向接口，正在取代传统的 CLI 和 SNMP。YANG 模型为路由器的各项功能提供了标准化的数据结构定义，NETCONF 则提供了基于模型进行配置和操作的协议框架。这使得上层平台可以自动化、程序化地完成对路由器的精细化配置和海量数据采集，为实现大规模网络的统一管控提供了技术基础。无论是下发一条用于阻断恶意宣告的 BGP 路由策略，还是订阅 ACL/FlowSpec 的匹配计数，这些标准接口都确保了平台能够对全网设备进行统一、实时且高效的管控。

4.2 全网流量的实时感知与处理

全网流量的实时感知与处理是图驱动与智能编排框架的“神经末梢”，贯穿于感知阶段的前端数据采集与预处理环节，核心目标是实现对网络中各类流量的全域、实时、多维度捕捉，并通过标准化处理

为后续的理解、决策阶段提供高质量数据输入。该环节打破传统网络中流量监测的碎片化局限，依托分布式采集节点与智能化处理引擎，构建覆盖“云、边、物” 全场景的流量感知体系，确保威胁特征、业务需求与网络状态的即时可见。

4.2.1 全域流量感知的核心维度

流量基础特征感知：通过 NetFlow、sFlow 等流采样技术，实时采集全网流量的五元组（源 / 目的 IP、端口、协议）、数据包大小分布、传输速率等基础属性。例如，对核心交换机的进出流量进行采样，精准识别流量突增、异常端口通信等潜在风险。

路由与安全状态关联感知：结合 BGP 路由更新消息、路由器接口状态（如 Up/Down）、安全模块运行日志（如 ACL 命中记录、DDoS 检测告警），将流量特征与网络拓扑、策略执行状态关联。例如，当某条 BGP 路由突然失效时，同步追踪该路由关联的业务流量是否出现路径切换或丢包，判断是网络故障还是恶意路由劫持。

应用层协议特征感知：解析 HTTP、DNS、SMB 等协议流量的负载内容，提取 URI 路径、域名请求模式、文件传输类型等应用层特征。例如，DNS 日志中出现 "xnjkwqer.random.com"这类熵值超过 7.5 的随机域名（正常域名熵值通常 <5），且每分钟发起 80 次递归查询，系统会自动匹配威胁情报库中的 DGA（域名生成算法）域名特征；对 HTTP 流量中 User-Agent 字段包含"Wget/1.16 (linux-gnu) --spider"且高频访问 /admin 路径的请求，结合 URI 中出现"../" 目录

穿越特征，判定为可疑漏洞扫描行为。

安全设备日志联动感知：汇聚 WAF、IPS、防火墙等安全产品的实时日志，提取攻击源 IP、攻击类型、防护动作等信息。例如，Cloudflare WAF 在 3 分钟内拦截同一 IP 的 15 次 CC 攻击（特征为不同 User-Agent 但相同请求频率的 HTTP GET），系统自动触发与 DDoS 清洗设备的联动，通过 GRE 隧道牵引该 IP 流量至清洗节点，同时从日志中提取攻击者指纹：地域、攻击工具、历史攻击记录，生成可视化的攻击者画像报告。

4.2.2 实时处理的关键技术机制

首先，机制的底层核心是其分布式数据采集架构。在边缘路由器、核心交换机、云边界网关等关键节点部署采集代理，通过“Master-Client”模式，可以在全网资产中部署轻量级的 Client 代理，构建了一个能够横向扩展的统一数据采集网络。这些客户端能够持续性地从主机和网络设备等多元实体中。具体的采集内容如图 4-2 所示。

汇聚的数据流被送至作为系统核心的 Master 总控平台，进行高性能的智能分析与深度处理。该平台整合了时序存储、分布式处理与并行计算技术，能够对海量数据进行高效的治理与挖掘。其处理能力可支持对高达 10Tbps 级别的网络流量进行瞬时、高保真的精准采样与分析。Master 平台不仅是数据存储库，更是一个多维度智能分析引擎，通过整合与关联来自不同维度的数据源，实现对网络健康状态的综合评估、异常行为的精准识别以及潜在安全威胁的深度洞察，最

终构建起一个立体化的实时网络监测体系。

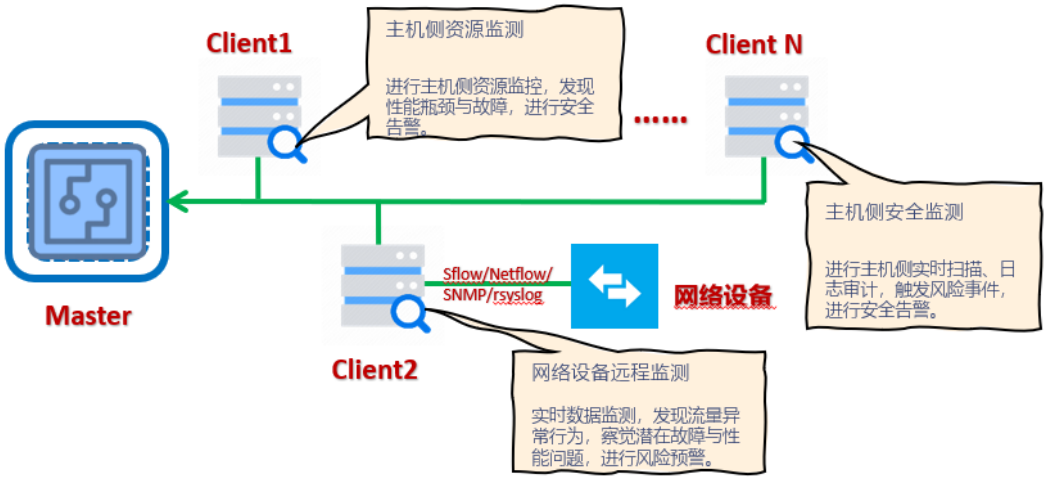

图 4-2 实时数据采集内容

4.3.3 与响应阶段的联动反馈

流量感知与处理环节并非单向数据输出，而是通过与响应阶段的实时联动形成闭环：响应阶段执行的路由策略调整（如流量重定向、带宽限制）会实时反馈至感知层，触发流量基线的动态更新。例如，当响应阶段对某攻击源执行带宽限速后，感知层会立即更新该源 IP 的流量基线，避免将限速后的正常流量误判为异常；同时，通过监测重定向后流量的清洗效果（如攻击包占比下降），验证响应措施的有效性，为策略优化提供数据支撑。

4.3 拓扑、流量与安全状态的统一图建模

拓扑、流量与安全状态的统一图建模是图驱动与智能编排框架中“理解阶段”的核心技术支撑，其核心目标是将分散的网络拓扑信息、动态流量特征与安全状态数据抽象为“节点-边”的图结构，此过程

是一个并行的、由数据驱动的持续性工作流，它将原始的感知数据实时地注入、实例化并关联到图模型中。

4.3.1 多源数据融合图结构

统一图建模以有向无环图（Directed Acyclic Graph，DAG）为基础，将网络拓扑、流量特征与安全状态抽象为图中节点与边的关联关系**节点**包含三类核心节点，分别对应拓扑节点、流量节点与安全状态和安全状态节点。具体结构如图 4-3 所示。

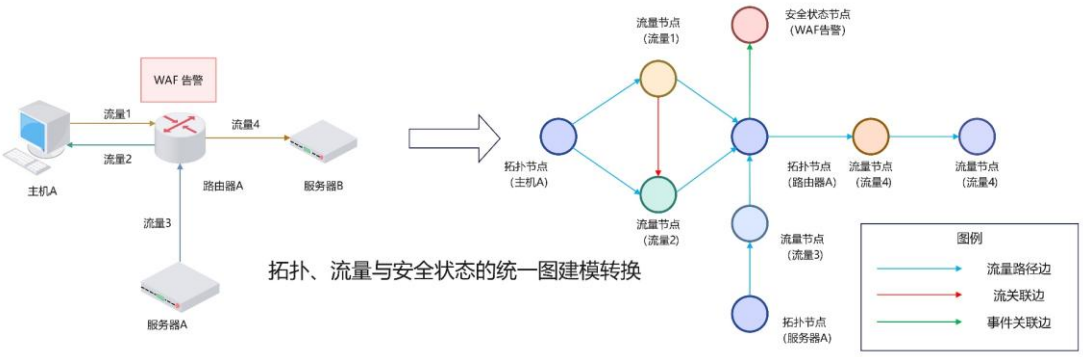

图 4-3 多源数据融合图结构

- 拓扑节点：代表网络中的物理或逻辑基础设施，如路由器、交换机、服务器、物联网设备等 。其节点属性主要为相对静态的配置信息，包含 IP/MAC 地址、设备类型、接口状态(Up/Down)以及通过 LLDP 或路由协议发现的拓扑层级关系

- 流量节点：代表一次具体的通信行为，是对一个或一组聚合后通信流的抽象 。其属性包括通过 NetFlow 提取的五元组（源 / 目的 IP、端口、协议）、数据包统计特征（如字节数、包长分布）及载荷二

进制特征向量，形成流量模态节点。

- 安全状态节点：代表一个具体安全事件的精细化建模。此类节点通常由外部安全系统事件实例化，如整合 WAF 拦截日志、IPS 告警、安全模块运行日志等安全事件作为一个独立的事件节点进行实例化。

边的定义体现数据间的依赖关系：拓扑节点与流量节点通过流量路径边关联，拓扑节点或流量节点与安全状态节点通过事件关联边连接（如异常 DNS 请求指向恶意域名节点），流量节点与流量节点之间通过流关联边进行连接。

- 流量路径边：核心功能是将逻辑上的通信行为映射到物理或虚拟的网络基础设施之上。此种边主要用于连接一个“流量节点”与一个或多个“拓扑节点”。当一个新的流量节点被实例化时，系统会解析其五元组信息中的源和目的 IP 地址，并结合图中已有的拓扑信息（如路由表、链路状态）来推算出该流量经过的转发路径。随后，系统会在该流量节点与路径上每一个关键的拓扑节点（如核心交换机、路由器）之间创建一条有向边。这条边清晰地表明了“此流量流经此设备”，从而为网络故障排查、流量工程以及基于路径的攻击溯源提供了直观的拓扑上下文。

- 事件关联边：扮演着将抽象安全事件与具体网络资产进行归属的关键角色。它主要连接一个“拓扑节点”或“流量节点”到一个“安全事件节点”。当一个源自 WAF、IPS 或 EDR 的日志被创建为一个安全事件节点时，关联引擎会立即解析该事件涉及的主体信息（如源 IP、

目的主机名等）。引擎会在图中查找与该信息匹配的拓扑节点或流量节点，并在二者之间建立一条“事件关联边”。例如，一个记录了 IP 地址 10.1.1.5 遭受 SQL 注入攻击的安全事件节点，会被一条边连接到图中代表 10.1.1.5 的服务器拓扑节点。这种关联使得安全告警不再是孤立的记录，而是直接附加到受影响资产上的、可供分析的动态属性，极大地提升了安全事件的上下文理解和响应效率。

- 流关联边：模型中最具分析深度的一种边，其设计旨在揭示不同通信行为之间隐藏的、非直接的内在联系，这对于发现如僵尸网络、分布式扫描、多阶段攻击等协同性威胁至关重要。与前两者不同，此种边仅在两个“流量节点”之间建立。其创建过程遵循一套严谨的规则，这些规则主要借鉴了流量拓扑分析的研究成果。规则主要包括：
  - 共同源关联：若两个流量节点拥有相同的源 IP 地址，则在它们之间建立一条关联边。这种关联有助于识别同一源头发起的批量、发散式通信行为，如 P2P 应用的数据广播或恶意扫描活动。
  - 通信链关联：若流量节点 A 的目的 IP 恰好是流量节点 B 的源 IP，则建立一条由 A 指向 B 的有向边。这种边能够有效地刻画出网络通信的接力或转发过程，对于追踪多跳攻击路径或服务调用链具有重要意义。
  - 时间邻近约束：上述所有流关联边的建立，都必须通过一个关键的时间窗口过滤器。只有当两个流量节点的产生时间戳之差小于一个预设的阈值 T（例如 3 秒 ），它们之间的关联才被认为是有效的。这一约束至关重要，因为它能确保所建立的关联

具有强时效性，有效排除了因时间跨度过大而产生的伪关系，并能显著控制图的边密度，从而提升后续图分析算法的性能与准确性。

4.3.2 图模型的动态构建过程

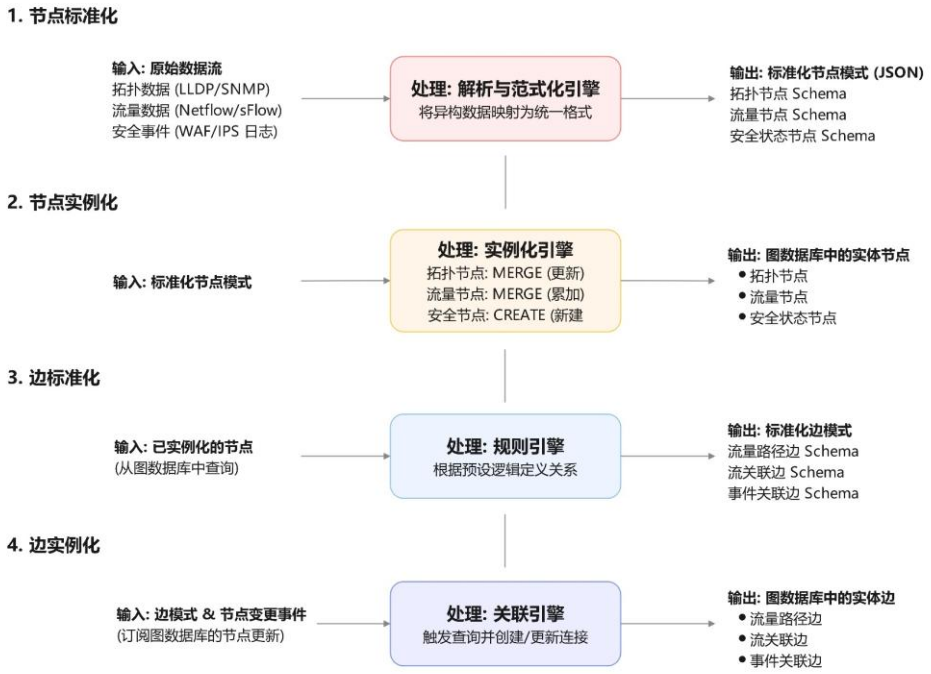

图 4- 4 图模型的动态构建过程

图模型的动态构建是一个由数据驱动的四阶段流水线过程，它首先通过标准化阶段将来自网络设备、流量探针和安全组件的异构原始数据流，解析并统一映射为拓扑、流量、安全状态这三类结构一致的节点模式，并依据关联逻辑定义了流量路径、流关联、事件关联这三类边的连接模式；随后，在实例化阶段，一个自动化引擎持续地将这些标准化的模式注入图数据库，通过对拓扑资产执行更新、对流量行

为进行累加、对安全事件进行独立创建的差异化策略来生成实体节点，并依据节点变更触发的规则，实时地创建或更新节点间的关系连线，最终将孤立的数据点实时编织成一张动态演进、关系丰富的全局网络图谱。具体构建过程如图 4-4 所示。

（1）节点标准化

图模型的动态构建始于节点标准化阶段，其核心目标是将异构、多源的原始数据流，转换为三种类型化、结构一致的节点模式（Node Schema）。此过程由一个多路数据解析与范式化引擎并行驱动。

拓扑节点模式:该模式的数据源主要为网络管理协议的输出，如 BGP 路由更新、LLDP 邻居发现报文及 SNMP MIB 轮询数据。解析引擎针对这些协议的特定格式进行解码，提取设备标识、接口状态及连接关系等信息。最终，所有信息被映射为一个标准的拓扑节点 JSON 对象，其核心字段包括：ip_address (IP 地址)、mac_address (MAC 地址)、device_type (设备类型)、interface_status (各接口 Up/Down 状态)以及通过 LLDP/BGP 解析出的 topology_hierarchy (拓扑层级关系)。

流量节点模式:该模式主要处理由采集的 NetFlow、sFlow 或 IPFIX 等二进制遥测数据。专用的解码器依据协议模板，将原始二进制流还原为结构化的通信记录。该记录随后被范式化为一个标准的流量节点 JSON 对象，其属性严格对应一次通信行为的抽象，包含：由源/目的 IP、端口、协议构成的 five_tuple (五元组)；由数据包统计（如字节数、包长分布、包间时延）计算得出的 statistical_features (统计特征向量)；以及（在深度包检测启用时）对载荷进行分析后生成的

payload_feature_vector (载荷二进制特征向量)。

安全状态节点模式 ：该模式专注于对安全事件的精细化建模，其数据源为 WAF 拦截日志、IPS 告警、EDR 检测日志等。日志解析器运用正则表达式或 CEF/LEEF 等标准格式解析库，从文本日志中提取事件元数据。这些元数据被统一转换为一个标准的安全状态节点 JSON 对象，用以封装一次独立的安全事件。其核心属性包括：event_source (事件来源)、event_type (事件类型)、severity (严重等级)、timestamp_event (事件时间戳) 以及包含原始日志和关键实体的 event_details (事件详情)。

通过此阶段，三种不同模态的数据被统一为三种定义清晰、结构固定的 JSON 模式，为后续的实例化流程提供了确定性的数据基础。

（2）节点实例化

在数据完成标准化映射后，这些规整的 JSON 对象将进入节点实例化阶段。一个高吞吐的消息队列会持续接收这些对象，并将其分发给一组按节点类型划分的并行工作进程（Worker Processes），以执行针对性的数据库事务。

拓扑节点实例化:由于拓扑节点代表相对静态的物理或逻辑资产，其实例化 ID 通常采用设备的 MAC 地址或主机名等稳定标识符(topo-{mac_address})。工作进程向图数据库发起的 MERGE（合并）查询，主要执行更新操作。例如，当接收到新的 SNMP 数据时，它会以“时间戳优先”的策略覆盖更新节点的 interface_status 属性；当接收到 LLDP 报文时，它会向节点的 topology_hierarchy 属性中追加或更新邻

居信息。

流量节点实例化:流量节点代表一次具体的通信行为，其实例化 ID 通过对五元组和聚合时间窗口进行哈希生成（flow-{hash(five_tuple)}-{time_window}），以支持对流的聚合。其 MERGE 查询逻辑是条件性的：如果具有相同 ID 的节点已存在，则执行“原子性加法”来累积 statistical_features 中的字节数与包计数值，并重新计算分布特征；如果不存在，则创建一个新的流量节点，并将当前 JSON 对象中的所有属性作为其初始值写入。

安全状态节点实例化:安全状态节点代表一个独立的、已发生的安全事件，具有不可变性。其实例化 ID 直接取自源安全系统的事件 UUID 或对原始日志的哈希（event-{source_uuid}）。因此，其数据库事务几乎总是 CREATE（创建）操作。工作进程为每一条告警日志创建一个全新的、独立的事件节点，确保每个告警在图模型中都有一个唯一的、不被后续数据覆盖的实体代表，从而保留了安全事件的完整性和原始性。

通过这一系列类型化、差异化的实例化策略，原始的感知数据被高效、准确地转化为图中持久化、可查询的实体节点，并确保了各类节点属性的动态更新符合其内在的数据逻辑。

（3）边标准化

在图中所有节点完成初步实例化之后，图模型构建流程进入边标准化阶段，其核心任务是为不同维度的数据关联关系定义统一、规范的连接模式。此阶段并非直接处理原始数据流，而是以图中已存在的

标准化节点为输入，通过一个规则引擎进行驱动。该引擎根据预设的关联逻辑，为三类核心关系——流量路径、流关联与事件关联——分别定义了标准化的边模式。

流量路径边模式：此模式定义了“通信行为”与“网络设备”间的归属关系。其生成规则被设定为：匹配一个流量节点的五元组属性中的源或目的 IP 地址与一个拓扑节点的 IP 地址属性。该模式规定了边的方向性和强制属性，从而为所有“流量途经设备”的场景建立了统一的数据结构。

流关联边模式：此模式旨在揭示不同通信行为间的内在逻辑。其规则集借鉴了流量画像分析理论，主要包括“共同源/目的关联”与“通信链关联”。例如，“通信链”规则定义为：当流量节点 A 的目的 IP 与流量节点 B 的源 IP 严格相等，且二者时间戳之差小于预设阈值 T 时，则满足关联条件。该模式确保了所有跨流量的分析型连接都遵循一致的判定标准和时间约束。

事件关联边模式：此模式用于连接一个抽象的“安全事件”与一个具体的“网络实体”（拓扑节点或流量节点）。其规则通过解析安全状态节点的事件详情触发，例如，提取 WAF 告警中的攻击源 IP，并将其与图中对应 IP 的拓扑节点或流量节点进行匹配。该模式的标准化在于，它将所有源自异构安全系统的告警，都统一转换为一种“事件-实体”的指向性关联。

此阶段的最终产出是一套抽象的、机器可读的边定义集合。每个定义都清晰地描述了一种关系的判定逻辑、源/目标节点类型以及必

要属性，为后续自动化、规模化的边实例化流程提供了结构化的蓝图。

（4）边实例化

边实例化的核心是将节点间的潜在关系，依据标准化模式，显式化、持久化为图数据库中的结构化连接。这是一个由关联引擎驱动的、持续运行的异步工作流。该引擎通过订阅图数据库的节点变更事件来触发操作，确保图的连接性随数据注入而动态演进。

当一个新的节点被实例化或其关键属性被更新时，关联引擎会启动一系列并行的图查询事务。以一个新创建的“流量节点-A”为例：触发路径关联：引擎立即发起一个查询，在所有“拓扑节点”中寻找其 ip_address 属性包含“流量节点-A”的源 IP 的节点。一旦匹配成功，引擎便会生成一个基于流量路径边模式的实例，并向数据库提交一个 CREATE 请求，建立一条从“流量节点-A”到“主机-X”的有向边。

触发流关联：同时，引擎会执行另一项查询，在图中检索与“流量节点-A”共享同一源 IP 且时间戳在 T 秒内的其他流量节点。对于每一个检索到的“流量节点-B”，引擎都会实例化一个基于流关联边模式的连接，其 relation_type 属性被赋值为“共同源”，随后创建这条双向或有向的边。

触发事件关联：反之，当一个“安全状态节点-S”（如 SQL 注入告警）被创建时，引擎会解析其 event_details，提取出攻击目标 IP。随后，它会查询图中所有与该 IP 相关的拓扑节点和近期活跃的流量节点，并根据预设的置信度算法，选择最相关的节点（例如“服务器-

Y”），最终实例化一条从“安全状态节点-S”指向“服务器-Y”的事件关联边。

在实例化过程中，为保证数据一致性与查询效率，每条边同样会生成一个基于其源/目标 ID 和类型的唯一哈希 ID。数据库操作普遍采用 MERGE 语义：若具有相同 ID 的边已存在，则仅更新其属性；若不存在，则创建新边。通过这一系列自动化的事务操作，原本孤立的数据点被实时地编织成一张动态演进、关系丰富的全局网络图谱。

4.4 可编排智能引擎

可编排智能引擎的核心目标是将图模型的推理结果转化为可执行的自动化流程，并根据网络状态变化动态调整策略逻辑，实现全流程智能化。该引擎的实现，依赖于分布式智能编排框架。通过动态的、可根据安全需求进行灵活编排的逻辑中枢，能够将统一图模型中的海量数据，转化为可行动的、实时的网络安全能力。

4.4.1 智能编排框架组成

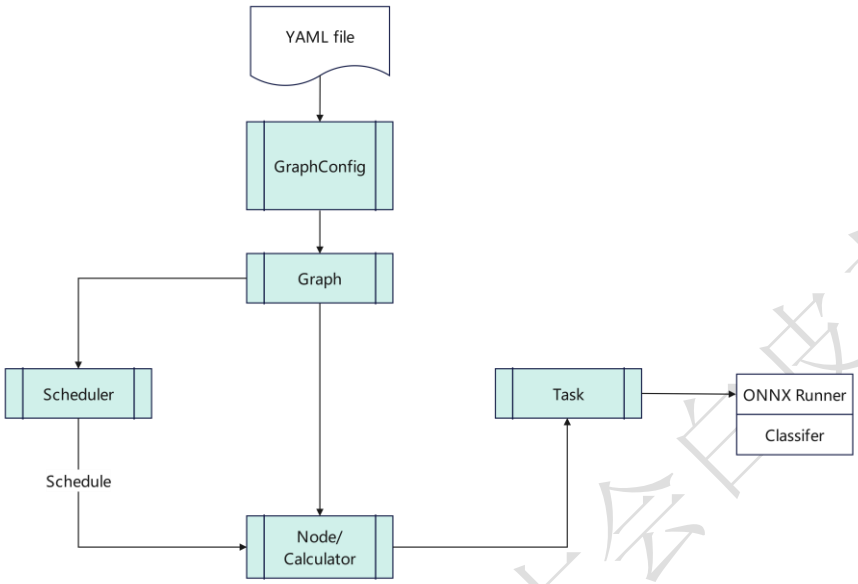

图 4-5 智能编排框架组成

（1）GraphConfig

GraphConfig 负责描述整个流水线的结构与配置，包括哪些计算器（Calculator）需要被实例化、它们之间的数据流连接方式，以及输入输出如何映射到外部资源等。在智能编排框架中，GraphConfig 通过解析 输入配置文件（pbtxt、YAML、JSON）来获取所需的配置信息，随后会将这些信息提供给 Graph 对象，用于构建完整的有向图模型。借助 GraphConfig，开发者可以方便地在框架中添加、移除或替换不同的计算节点，灵活地对数据流进行重定向，从而实现对异常流量检测与分类流程的可扩展管理。

（2） Graph

Graph 是根据 GraphConfig 创建并运行整个数据流图的核心实体。它会读取并解析 GraphConfig 中的节点定义及流连接信息，将各个 Calculator（包括自定义的 ONNXRunner、Classifier 等）组装成一个有向图。Graph 在执行时，会自动管理节点之间的数据流动及并行执行顺序。通过对数据包（Packet）进行时间戳管理，Graph 可以在时间维度上协调各节点的处理流程，以保证异常流量检测场景下的时延与准确度。

（3）Scheduler

调度器（Scheduler）负责动态管理图中节点的执行顺序和资源分配。调度器基于数据依赖关系和节点的实时状态（如输入流的满足情况）决定节点的执行时机，而非固定优先级规则。所有节点的任务通过一个全局线程池分配，线程数量根据硬件能力自动调整。调度器确保高效利用系统资源，同时提供灵活的任务优先级配置，允许开发者为关键节点分配更多资源。在智能编排框架中，调度器进一步扩展，支持复杂的多线程环境，通过动态调整任务优先级和节点分组机制，确保流量分类等关键任务在高负载场景下的实时性和稳定性。

（4）Node/Calculator

节点（Node）是执行数据处理的核心组件，每个节点实现为一个独立的计算器（Calculator），负责接收输入流或旁路数据包，进行处理后将结果传递到下游节点。源节点通常从外部读取数据流（如文件或网络流量），而非源节点则通过输入策略（如时间戳匹配）确定执行条件。框架保证节点的线程安全性，使每个节点在单线程中运行，

从而避免数据竞争问题。在智能编排框架中，节点通过模块化设计实现预处理、特征提取和分类等功能，开发者可灵活替换或扩展节点，以适应不同的异常流量检测需求。模块化和灵活性使得框架能够快速适配新的任务，同时确保框架整体的高效性。

（5）Task

在智能编排框架中，ONNX Runner 和 Classifier 都是对 Task 接口的具体实现，分别用于载入已训练模型和流量分类。通过继承和扩展 Task 接口，开发者可以插入自定义的业务逻辑模块，满足在智能编排框架中对预处理、特征提取和模型推理等功能的需求，同时保持框架设计的模块化和扩展性。

4.4.2 核心功能

（1）弹性资源管理

弹性资源管理旨在根据工作负载的实际需求，动态地分配和调整资源，包括计算资源（如 CPU、内存、GPU）、存储资源以及网络资源等。它能够在工作负载增加时自动分配更多资源以保障服务性能，在工作负载减少时释放闲置资源，避免资源浪费。通过将集群资源按照组织架构进行分层，形成资源池，并以树形结构呈现，具体结构图如图 4-6 所示。根节点（Root）下有组织节点（Org），组织节点下又细分团队节点（Team）。每个层级的资源池都关联一组资源配置参数，包括资源预留（Reservation，R）、权益资源（Entitlement，E）、共享资源（Share，S）以及资源上限（Limit，

L）。资源预留是为该层级强制保障的最小资源量，权益资源是默认应得的资源比例，共享资源可在层级间弹性借用，提高资源利用率，资源上限则用于防止单个层级过度占用资源。

分布式计算引擎具备动态调整资源分配的能力。在 Kubernetes 集群中，当某个工作负载的资源需求发生变化时，分布式计算引擎可以实时感知并快速重新分配资源。例如，在机器学习训练中，随着训练数据量和模型复杂度的动态变化，分布式计算引擎能够为训练任务及时调配更多或释放多余的 CPU、GPU、内存等资源，相比 K8s 原生的资源管理方式，能更高效地利用集群资源，避免资源闲置或浪费。

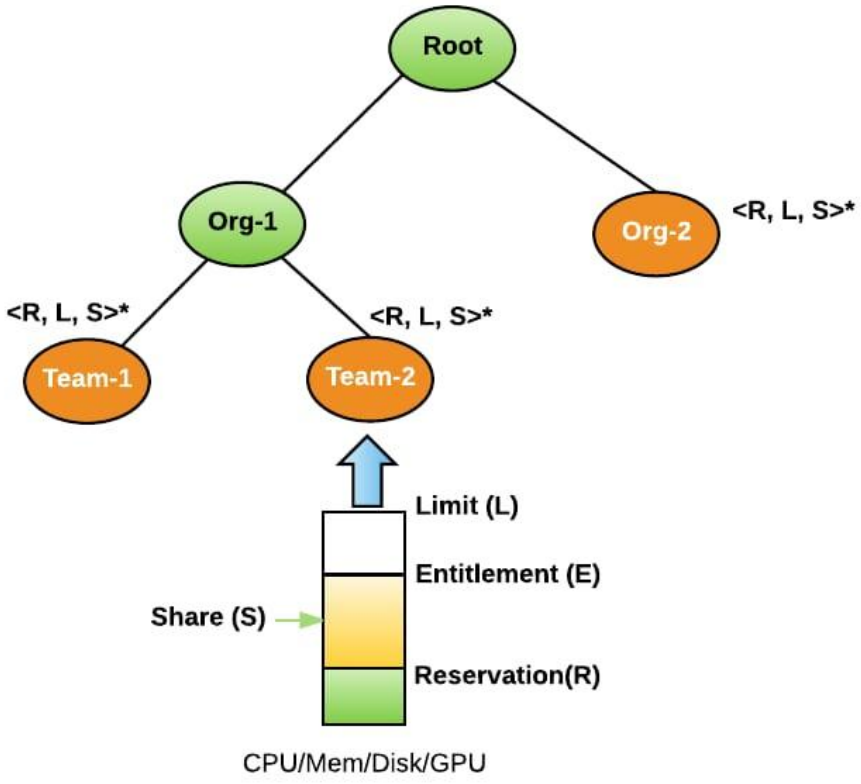

图 4-6 弹性资源管理结构图

（2）异构集群支持

在包含不同类型硬件（如 CPU 和 GPU）的混合集群中，能够有效运行训练作业。一方面，通过将不需要 GPU 的任务卸载到 CPU 节点，实现资源的合理利用，比如在机器学习训练中，将数据加载和混洗等任务放在 CPU 节点处理，再将处理后的数据传输到 GPU 节点进行模型训练。另一方面，开发 GPU 过滤插件，让非 GPU Pod 和 GPU Pod 分别在 CPU 节点和 GPU 节点上运行，并采用不同的调度策略，如负载感知策略用于 CPU 节点的 Pod 分配，装箱调度策略用于 GPU 节点的 Pod 分配。

（3）动态编排逻辑

引擎的动态编排逻辑体现在其执行过程并非一成不变，而是能够根据数据和中间结果进行自适应调整，这由其事件驱动的调度器（Scheduler）来实现。

该调度器基于数据依赖关系来管理任务的执行，而非固定的时间线或优先级。这意味着，一个计算节点的执行，是由其所有上游输入数据全部“准备就绪”这一事件来驱动的。这种机制天然地支持了动态和并行的工作流。例如，当原始数据进入后，两个并行的特征提取节点会同时开始工作。调度器会监控它们的状态，只有当其中一个节点（例如，提取统计特征的节点）率先完成后，它才会立即将结果数据传递给下游对应的推理节点并触发其执行，而无需等待另一个并行的

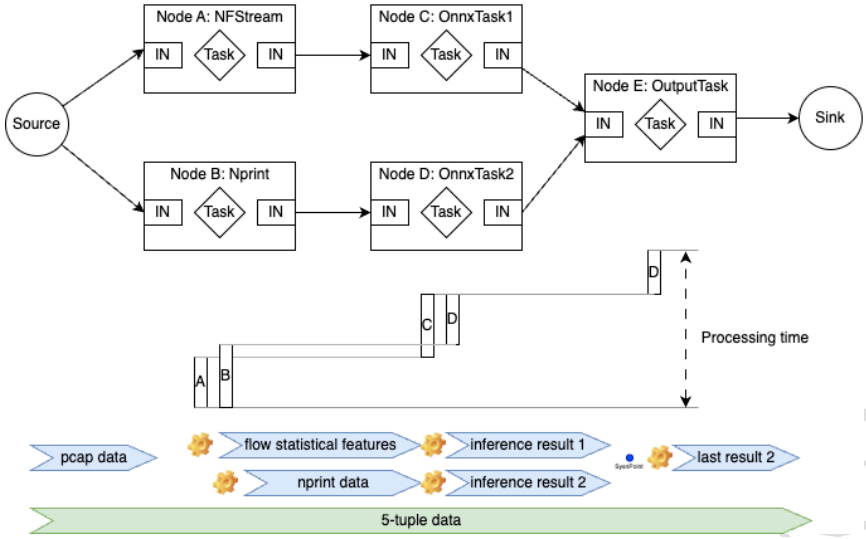

图 4- 7 并行特征提取和推理分析

特征提取任务结束。具体示例如图 4-7 所示。

更进一步，这种事件驱动的机制允许实现条件执行和逻辑分支。一个分析工作流的走向，可以由上一个节点的计算结果来动态决定。例如，可以设计一个“初步风险评估”节点，它会先对流量进行快速分类并输出一个风险评分。调度器可以根据这个评分结果，将流量动态地导向不同的处理路径：如果评分高于阈值，则将数据发送到一个需要消耗大量计算资源的“深度载荷分析”节点进行精细化检测；如果评分较低，则可能只将其发送到一个简单的“日志记录”节点。通过这种方式，引擎的分析逻辑能够实时地根据威胁的实际情况进行调整，将宝贵的计算资源集中在真正高风险的事件上，这就是其“动态逻辑”的核心体现。

4.5 插件化机制

4.5.1 ONNX 与模型模块化

ONNX 作为一项行业性的开放标准，其根本目标是解决机器学习领域中不同开发框架与部署环境之间的壁垒问题，为模型提供统一且中立的中间表示。这一标准的确立，是实现 AI 模型工程化与模块化的逻辑起点。在缺乏统一标准的情况下，模型与其训练框架、特定的运行时环境深度绑定，形成了紧耦合的技术孤岛，极大地阻碍了模型的复用、迁移与迭代。ONNX 通过定义一套标准的计算图结构、算子集合和文件格式，充当了模型生产者与模型消费者之间的“技术契约”，确保了只要遵循此规约，模型便能脱离其原始开发环境，作为一个独立的、可预测的单元而存在。

深入分析一个 ONNX 文件的内部结构，可以更清晰地理解其模块化设计。文件的核心是计算图协议，它容纳了模型的所有构成元素。首先，计算图的公共接口由其输入和输出字段严格定义，每个接口都详细描述了张量的名称、数据类型及维度信息，这构成了模块清晰的外部边界。其次，图的内部实现由一系列节点构成，每个节点都是一个标准算子的实例，并精确地指定了其输入输出关系，共同组成一个有向无环图来描述数据处理的全过程。至关重要的一点是，模型的所有已训练参数，如卷积核的权重、全连接层的偏置等，都通过初始化器被序列化并包含在文件之内，这使得 ONNX 文件成为一个自包含的模块，无需依赖外部文件即可完整地重建模型状态。

基于上述特性，ONNX 模型在现代 MLOps 体系中扮演了关键的模块化角色。其兼容性与生命周期由算子集版本提供保障。每一个 ONNX 模型都声明了其依赖的 opset 版本，而推理引擎则依据此版本来确保对模型中所有算子的正确支持，这为模块的版本迭代与向后兼容提供了可靠依据。因此，一个经过验证的 ONNX 模型可以被视为一个稳定的软件构件，能够被存储在构件仓库中进行版本化管理，并通过 CI/CD 流水线被独立地部署到任何支持其 opset 版本的云端或边缘设备上。这种标准化的封装与管理方式，正是将 AI 模型从研究原型转化为健壮、可靠的工程模块的核心所在。

4.5.2 自定义扩展与编排模块化

虽然 ONNX 标准提供了丰富的算子集，但在实际应用中，为了实现差异化的业务逻辑或极致的性能优化，仅依赖标准算子往往是不够的。此时，就需要通过自定义扩展机制来增强系统的能力，而自定义算子是实现原子功能模块化的关键手段。当需要引入专有算法、标准库未覆盖的数据处理逻辑、或针对特定硬件（如 FPGA、ASIC）的计算核时，开发者可以创建自定义算子。此过程遵循严格的模块化设计：算子需被定义在唯一的领域标识之下以避免命名冲突；其核心计算逻辑通常采用 C++或 CUDA 等高性能语言实现，并被编译成独立于模型的动态链接库。推理引擎在运行时，通过指定的 API（如 ONNX Runtime 的 register_custom_ops_library）动态加载这类库，从而使新的运算能力对当前会话可用。这种机制将算子的实现与模型本身、与

推理引擎核心都进行了解耦，使算子库成为一个可被多个模型共享、可独立升级和分发的功能模块。

在原子化的功能扩展之外，模块化的思想也体现在更高层次的逻辑组合与抽象上，这主要通过 ONNX 标准中的函数机制来实现。该机制允许开发者将计算图中的一系列基础算子子图构成，并封装成一个可复用的、更高阶的新算子。例如，一个包含多头自注意力、残差连接和层归一化的 Transformer 编码器层，可以被完整地定义成一个函数。在主计算图中，可以直接像调用普通算子一样调用这个函数节点，而无需关心其内部复杂的实现细节。这种方式不仅极大地简化了主计算图的结构，提升了可读性与可维护性，也为推理引擎在执行时提供了更大的优化空间，因为引擎可以将整个函数作为一个整体进行调度或编译优化。它与自定义算子的关系在于，自定义算子是引入新的、基础的计算能力，而函数则是对已有的计算能力进行组合与封装。

这两种扩展机制的结合，最终为上层的流程编排系统提供了极大的灵活性，使其能够实现真正的编排模块化。编排系统现在可以调度和组合三种不同粒度的模块：代表完整业务流程的 ONNX 模型、提供原子功能的自定义算子库、以及包含抽象逻辑组合的 ONNX 函数。一个复杂的 AI 工作流可以被清晰地分解和构建，例如，可编排智能引擎首先调用一个预处理模块，该模块使用了一个加载自 lib_preproc.so 的自定义算子来执行特殊的数据增强；其输出接着被送入一个大型的、使用标准算子的目标检测模型；最后，检测结果被传递给一个后处理模块，该模块内部调用了一个计算图协议定义的复杂

非极大值抑制（Non-Maximum Suppression，NMS）函数来筛选检测框。在整个过程中，可编排智能引擎负责管理模块间的数据依赖与流转，并确保每个阶段所需的扩展库都已被正确加载。这充分体现了通过精细化的模块分解与组合，构建复杂、健壮且易于演进的 AI 应用的能力。

五、框架落地与场景实践

5.1 全网 DDoS 攻击检测与缓解方案

5.1.1 案例背景

在数字化浪潮席卷全球的今天，分布式拒绝服务（Distributed Denial of Service，DDoS）攻击已演变为网络空间中最具破坏力、最常见的安全威胁之一。攻击的规模从 Gbps 级别跃升至 Tbps 级别，攻击手法也从单一的容量耗尽型攻击，演变为包含应用层攻击、脉冲式攻击、“低慢速”攻击在内的复杂混合型攻击。这种演进趋势对所有依赖网络提供服务的组织构成了严峻挑战，传统的 DDoS 防御方案在应对现代高级威胁时，其固有的局限性日益凸显。

尽管市场上存在多种 DDoS 检测与缓解方案，但许多现有方案在设计理念和技术实现上仍存在明显的不足之处，主要体现在以下几个方面：

检测视角的局限性：传统的 DDoS 防御体系通常采用单点部署模

式，例如仅在数据中心入口或互联网出口部署检测设备。这种“管中窥豹”式的检测方式，缺乏对全网流量拓扑和时空特征的宏观洞察力。它或许能发现指向某一特定目标的攻击流量，但无法有效还原攻击在整个网络中的传播路径、影响范围以及潜在的溯源线索。当攻击者采用多点、分散的攻击源时，这种局部视角极易造成判断失误，难以形成全局性的、协同一致的防御策略。

检测逻辑的滞后性：许多现有方案的核心检测逻辑仍然基于静态的流量阈值或固定的攻击特征规则。这种“一刀切”的方法在面对流量平稳、模式简单的网络环境时或许尚能应付，但在业务流量复杂多变、攻击手法不断翻新的今天则显得力不从心。对于“低慢速”攻击、加密流量攻击以及模拟合法用户行为的应用层攻击，静态规则往往会产生大量的误报和漏报。它无法建立动态的、与业务紧密结合的流量基线，更不用说利用深度学习等智能技术去识别那些隐藏在海量正常通信中的细微异常模式。

响应机制的割裂性：在众多防御体系中，攻击的“智能检测系统”与“缓解响应系统”（如流量清洗设备）往往是两套独立的系统。当检测系统发现攻击后，通常只能生成告警，需要安全运维人员介入分析，再手动配置清洗策略或引流策略。这一过程不仅耗费宝贵的人力资源，更重要的是，在检测和缓解之间造成了数分钟甚至更长的“响应延迟”。在 DDoS 攻击分秒必争的战场上，这个延迟的“窗口期”足以让业务中断，造成不可挽回的损失，防御效果大打折扣。

适应能力的匮乏性： 随着物联网（IoT）设备的普及，新型僵尸网

络的规模和复杂性空前增长。同时，网络的带宽和复杂度也在持续提升。传统的 DDoS 防御设备在架构上可能难以扩展，无法满足大规模网络的性能需求。更重要的是，其固化的检测模型和功能更新缓慢，面对层出不穷的新型攻击载体和技术，常常显得“捉襟见肘”，缺乏持续自适应学习和演进的能力。

综上所述，一个缺乏全网视角、依赖静态逻辑、响应流程割裂且适应性差的防御体系，已无法有效应对当前复杂、智能的 DDoS 威胁格局。因此，业界迫切需要一种新一代的智能防御方案，它必须能够实现对全网流量的全面、快速、精确识别，并指导形成协同、高效的自动化防御闭环。

5.2.2 解决方案

针对上述背景中提到的传统 DDoS 防御方案的种种局限，我们基于网络原生智能框架，设计并部署了全网智能 DDoS 检测与协同防御体系。该方案从根本上摒弃了网络与安全相互割裂的传统模式，将全网流量的精准检测、攻击行为的智能决策与阻断等自动化缓解手段深度融合，实现了从“秒级检测”到“秒级响应”的全流程自动化闭环。

整体架构如下图所示，其核心逻辑是：通过遍布全网的采集点实时捕获流量数据，由智能分析引擎进行深度学习与行为建模分析，一旦识别攻击，可通过安全网络数据智能平台协同联动网络中智能分析引擎，自动执行流量精准阻断等防御策略并下发到可编程交换机进行 DDoS 攻击缓解。

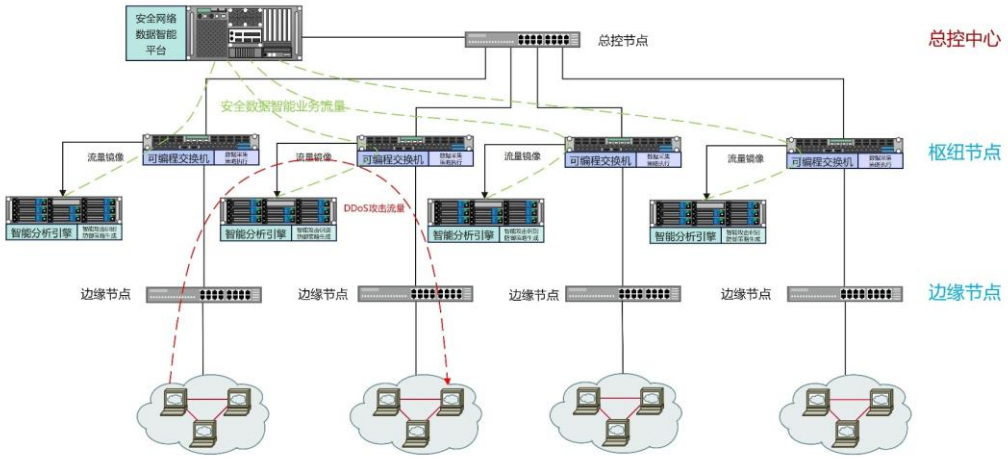

图 5-1 全网 DDoS 攻击检测与缓解方案示意图

（1）数据采集

为实现对网络核心流量的全面洞察，方案采用了一种高保真、非侵入式的数据采集策略。针对可编程交换机这类采用专用芯片进行高速转发的核心网络硬件，可编程交换机通过其旁路镜像功能进行数据采集。该技术将所有流经可编程交换机的实时流量，完整地复制一份，并旁路发送至一台专用的分析服务器作为智能分析引擎。部署在该服务器上的轻量级分析代理负责接收并处理这些海量的镜像数据，进行后续的抽样与分析。这种方式的优势在于对可编程交换机本身的转发性能做到零侵扰，在不影响主干网络正常运行的前提下，获取了最原始、最完整的流量全貌，为后续的智能分析与精准决策提供了坚实的数据基础。

（2）多维数据驱动的智能攻击识别理解

在完成全面数据感知后，汇聚而来的数据将注入系统的“大脑”——智能分析引擎，进行深度理解与攻击识别。该引擎运用先进的行为

分析算法，对流量进行精细化解构，能够从协议类型、报文长度、源端口随机性等多个维度精准刻画攻击特征。在混合了正常业务的复杂场景下，智能分析引擎更能体现其智能性，它通过关联分析流量模式与设备性能指标，能够准确地将恶意攻击从海量背景流量中剥离出来，显著降低了传统方案的误报与漏报率。这一阶段的核心任务，是将纷繁复杂的原始数据转化为清晰、准确、可操作的攻击事件情报。

（3）自动化、精准的防御策略生成

一旦智能分析引擎确认了攻击事件，智能分析引擎便无缝衔接到决策阶段，自动生成高度精准且可解释的防御策略。这些策略并非宽泛的封堵指令，而是包含了明确五元组信息和处置动作的精细化规则。系统的决策能力足以应对大规模、分布式的复杂攻击。即便面对同时攻击数百个不同目标的场景，智能分析引擎依然能为每一个被攻击 IP 独立生成并下发对应的防御策略，实现“点对点”的精确保护。这个自动化、智能化的决策过程，是连接威胁情报与有效防御之间的关键桥梁。

（4）端到端、闭环化的协同联动响应

流程的最后一步是将决策转化为行动，通过自动化的协同响应机制，完成对威胁的闭环处置。智能分析引擎生成的防御策略被设计为可直接下发至可编程交换机，并自动转化为标准的访问控制列表等设备可执行的规则。这些规则一旦生效，便会立刻对匹配攻击特征的恶意流量进行实时过滤与阻断。更重要的是，这是一个迭代式的防御过程。当最主要的攻击流量被阻断后，原先被掩盖的次要攻击会暴露出

来，随即被系统在新的检测周期中捕获并清除，从而实现对攻击流量的深度、持续性清洗，确保了业务的连续性和网络的安全性。

5.2 路由安全一体化解决方案

5.2.1 案例背景

作为互联网的关键基础设施，域间路由系统安全是网络空间安全的重要基石。以 BGP 为基础协议的全球互联网经过 50 多年的蓬勃发展，逐步从计算机互联网、消费互联网向产业互联网演进，成为全社会数字化基础设施，因而对安全可信的路由服务诉求越来越强烈。作为互联网数据传输的核心，互联网不仅在数据转发性能方面，而且在拓扑结构、健壮性、安全性等方面也都高度依赖域间路由系统。域间路由系统对于整个互联网的可靠稳定运行具有重要意义。

传统的应对方案在面对控制平面威胁时，通常面临以下挑战：

响应机制滞后，缺乏时效性：安全事件的处置严重依赖网络工程师手动排查、定位问题、登录设备执行命令行进行策略封堵，整个过程耗时良久，无法在攻击发生的第一时间进行有效遏制。

系统能力割裂，缺乏联动：安全监控系统与网络管理系统相互独立，安全分析产生的告警无法自动转化为网络侧的防御策略，缺乏有效的闭环协同机制。

控制平面状态的可见性缺失：对全网的 BGP 路由状态缺乏全面、实时、精细的可见性，难以快速识别异常路由的来源和影响范围。

为了应对上述挑战，构建一个能够主动感知、智能决策、并自动处置路由威胁的现代化防御体系迫在眉睫。本案例将介绍一种基于“安全网络一体化”理念的创新解决方案

5.2.2 解决方案

针对上述背景中提到的 BGP 路由攻击场景，我们结合网络原生智能架构，设计并部署了一套以“安全网络一体化平台”为核心的闭环路由安全解决方案。该方案摒弃了传统安全与网络分离的模式，将威胁感知、智能分析与网络配置变更融为一体，实现了从“发现”到“处置”的全流程自动化。

整体架构如图 5-2，方案的核心逻辑是：通过标准化的数据采集通道实时监控路由器集群的控制平面状态，由平台进行智能分析决策，并经由标准化的安全配置通道实现对恶意流量的精准、自动化处置。

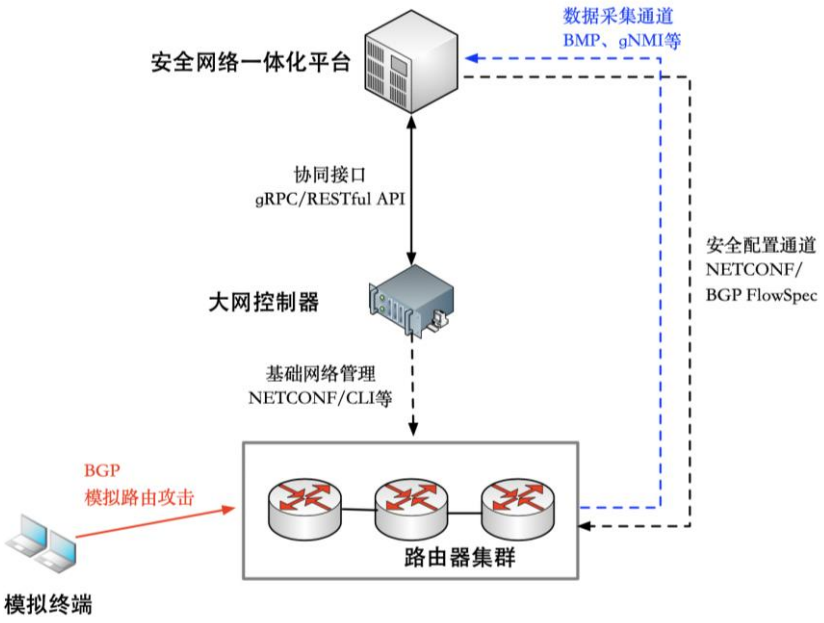

图 5-2 路由安全一体化解决方案示意图

具体实现步骤如下：

（1）实时感知：基于协议的深度数据采集

首先，模拟终端作为自动化测试与验证工具，按照预设方案向网络中发起一次可控的路由劫持攻击。当这条恶意的 BGP 路由更新报文抵达网络边缘的路由器集群时，路由器集群在根据 BGP 协议进行常规路由计算的同时，也立即履行其作为 “感知探针”的职责。它通过 BMP 协议，将这条包含了攻击特征的原始 BGP 更新报文，实时、无损地传送安全网络一体化平台。

（2）智能理解：基于可信基准的自动化检测

原始数据流抵达安全网络一体化平台后，便进入一条自动化的内部处理流水线。 数据采集与适配模块将其转换为平台内部的统一路由事件模型，并分发至核心分析模块与数据存储与管理模块。核心分析模块作为分析中枢，会立即执行多维度检测：它将该事件的源 AS、前缀等关键属性，与数据存储与管理模块中预设的 RPKI 源 AS 授权、ASPA 商业关系等可信基准进行交叉验证，最终将此事件精准地判定为一次“路由劫持”攻击，并生成结构化的告警。

（3）协同决策：融合网络上下文的策略生成

“路由劫持”的分析结论会立刻被送至决策与处置编排模块。为确保处置的精准性，该模块可调用与大网控制器的协同接口，查询受影响路由器的网络拓扑、设备角色等基础上下文信息，以丰富决策依据。在获得了完整的“安全告警 + 网络上下文”信息后，该模块才最终决策出最佳的处置方案，并自动生成一个协议无关的抽象处置指令，例

如一条用于精确丢弃恶意流量的 BGP FlowSpec 规则意图。

（4）闭环响应：基于标准化接口的自动化执行

编排好的抽象指令通过标准化的安全配置通道，被下发至作为执行单元的目标路由器集群。平台的南向通道适配器会将该指令翻译为具体的 NETCONF 配置或 BGP FlowSpec 宣告，并部署到设备。路由器接收到指令后，会即刻应用此安全策略，在硬件层面快速、精准地阻断由劫持路由所引入的非法流量，至此便完成了一次从攻击发生到威胁解除的自动化闭环处置。与此同时，整个处理过程的所有状态都会被实时汇聚到平台的管控与呈现模块，网络管理员可以通过图形化的 Web UI 清晰地监控此次安全事件的完整生命周期，实现了对路由安全的“可管、可控、可见”。

六、架构生态与未来展望

6.1 模块化开放的架构、生态与接口

ONNX 作为 AI 领域的开放标准，其模块化设计不仅体现在模型封装和扩展机制上，更延伸至系统级架构、生态建设和接口规范，共同构筑了一个前沿、协作的 AI 框架体系，随着 LLM、边缘计算和异构硬件加速的快速发展，ONNX 已演变为支持多模态 AI 和高效部署的核心枢纽。本小节将从模块化开放的架构、生态以及接口三个维度详细阐述 ONNX 如何实现 AI 系统的互联互通与持续创新，确保模

型在动态环境中无缝迁移、优化和扩展。

6.1.1 ONNX 的开放架构

ONNX 的架构设计遵循模块化开放原则，以计算图（Graph）为核心，构建了一个松耦合、可扩展的系统框架。这一架构将模型表示、执行引擎和硬件适配层解耦，允许开发者在不修改核心组件的情况下注入新功能。2025 年的最新发展中，ONNX v1.18.0 及后续迭代引入了对动态形状和量化支持的增强[30]，进一步适应了 LLM 和实时 AI 场景的需求。例如，通过 MLIR-based Compiler 的集成[31]，ONNX 架构现在支持更高效的中间表示编译，允许模型在编译时进行跨框架优化，减少了从训练到推理的转换开销。

在架构层面，ONNX 强调分层模块化：顶层是模型层，使用 Protobuf 序列化的 GraphProto 定义静态计算图，包括节点（NodeProto）、初始化器（TensorProto）和版本声明（OperatorSetIdProto），这确保了模型的自包含性和可移植性[32]。中层是运行时层，以 ONNX Runtime 为核心，支持插件化后端适配器（如 CPU、GPU、NPU），开发者可动态加载自定义执行提供者（Execution Providers），如 Qualcomm 的 QNN GPU backend[33]，实现针对 Adreno GPU 的硬件加速。底层是硬件抽象层，通过开放接口连接异构设备，支持从云端到边缘的部署。

6.1.2 ONNX 的生态体系

ONNX 的生态体系是一个由开源社区、框架提供商、硬件厂商和

企业用户共同构建的协作网络，2025 年已扩展至涵盖 LLM、边缘 AI 和多模态应用的全面链条[34]。作为 LF AI & Data 基金会的毕业项目，ONNX 采用 Apache 2.0 许可，促进全球贡献者参与，年会如 2025 ONNX Annual Meetup 展示了 steering Committee 的更新，包括对大型模型 IR 的增强支持。

在框架生态方面，ONNX 获得了广泛兼容：PyTorch 通过 torch.onnx.export 无缝导出模型，TensorFlow 集成 tf2onnx 转换器，其他如 MXNet、Scikit-learn 和 PaddlePaddle 也提供插件支持。推理侧，ONNX Runtime 作为枢纽，与 Azure ML、AWS SageMaker 和 Google Cloud AI 集成；硬件伙伴如 NVIDIA（TensorRT-ONNX）、Intel（OpenVINO）、Qualcomm（SNPE-ONNX）和 AMD 积极贡献优化模块[35]，确保模型在 GPU、NPU 上的高效运行。

6.1.3 ONNX 的接口机制

ONNX 的接口机制标准化了模块间交互，确保互操作性和扩展性。核心是 Protobuf 协议接口，用于模型序列化，包括 GraphProto 定义计算图、TensorProto 处理张量数据，以及 OperatorSetIdProto 管理版本。这些接口提供精确语义，支持任何工具解析 ONNX 文件，而无需自定义适配。

运行时接口以 ONNX Runtime API 为主，支持多语言（如 Python、C++、Java、C#），例如 SessionOptions 配置自定义算子加载，Run 方法标准化推理流程。扩展接口如 CustomOpApi 允许注册自定义算子，

ONNXIFI 提供后端集成规范，支持专有硬件加速。

6.2 迈向全面零信任及下一代 SASE 与 SD-WAN

在网络原生智能的驱动下，下一代 SD-WAN 已远超第一代产品优化连接和降低成本的范畴，演变为一个具备预测和自愈能力的智能网络平台[36]。

6.2.1 具备预测与自愈能力的下一代 SD-WAN

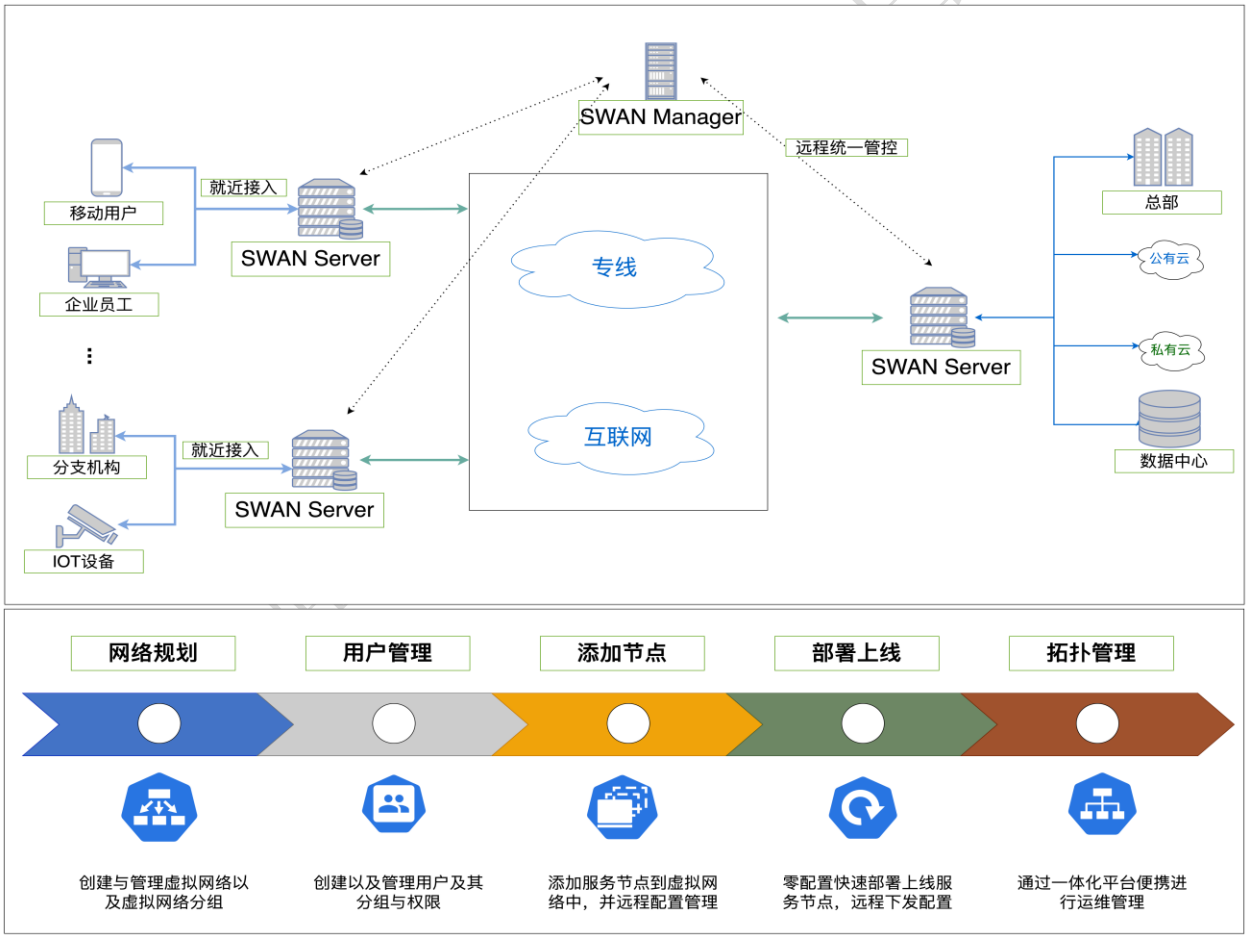

图 6-1 SWAN 组网结构和流程步骤

在网络原生智能的驱动下，下一代 SD-WAN 的使命已远超第一代产品优化连接和降低成本的范畴。它演变为一个具备认知能力的智能

网络平台，成为整个 SASE 架构坚实、敏锐的“神经网络系统”[37]。其核心的升级体现在三个层面：首先，它具备了从被动响应到主动保障的预测能力，能够预见并规避网络质量问题；其次，它实现了从简单故障切换到业务自愈的升华，能够在故障发生时进行智能化的路径重规划与策略自适应；最后，它完成了从执行静态规则到理解业务意图的转变，能够为关键应用做出自主的、以体验为中心的决策。

SD-WAN 的核心变革在于其预测能力，它通过在网络边缘部署轻量级探针并结合 AIOps 平台，对海量的遥测数据进行持续学习，从而将网络管理从被动响应转变为主动保障[38]。当网络出现故障时，下一代 SD-WAN 能够实现真正的网络自愈，其内涵远比传统的故障切换丰富。这一能力建立在对全网拓扑、业务策略和实时状态的全局视野之上[39]。例如，当某分支机构的核心路由器意外宕机，AIOps 平台能立即定位故障根源为硬件失效，并自主进行一次全局的路径重规划，可能会将高优先级的 ERP 流量引导至高质量 MPLS 链路，而将普通办公流量分流至多条互联网宽带，避免单点拥塞。同时，系统会自动将原路径关联的所有安全与 QoS 策略动态迁移并应用到新路径上，在毫秒级内完成业务恢复，最大限度地保障了业务的连续性。

此外，下一代 SD-WAN 的核心是基于应用意图的自主决策，彻底将网络管理从繁琐的微观配置中解放出来。平台能够深度识别上千种应用的“指纹”，并理解其对网络的独特需求。在应对某突发性国家级安全事件时，指挥中心的 IT 管理员不再需要手动配置复杂的 QoS 和路由策略，只需声明最高优先级的业务意图：“为国家指挥中心、一线

移动单位和无人机侦察图像回传之间，建立一条高带宽、低延迟、抗干扰的加密通信线路”。系统接收到此意图后，便会自主地编排网络资源，动态聚合 5G、卫星和专线链路，应用军工级加密标准，并强制征用网络带宽，确保指挥、控制和情报（C2I）数据流的绝对优先传输。

6.2.1 实现全面零信任的的下一代 SASE

如果说下一代 SD-WAN 是智能的“循环系统”，那么由网络原生智能驱动的下一代 SASE 就是智能的“免疫系统”[40]。它将零信任原则从一系列需要人工维护的静态配置规则，转变为一个动态的、能够自主执行并持续进化的安全能力。其基石是一个将网络拓扑、流量行为、用户身份、设备状态、应用漏洞及威胁情报等所有信息统一建模的全局安全知识图谱。基于此图谱，网络原生智能引擎能够进行深度推理，发现隐藏在海量数据中微弱的风险“信号”，从而将安全防御从被动响应提升至主动预测[41]。例如，通过发现“同一身份、多设备、异常行为”之间的隐藏关联，系统能预测横向移动攻击的早期侦察阶段；或是在新漏洞披露后，立即模拟并找出潜在的攻击路径，让防御者抢占先机。

这是对零信任“假设泄露”原则的有效实践，当威胁被识别或预测后，下一代 SASE 的响应是自动化的、闭环的，从而实现安全层面的自愈。这个“从检测到阻断的原生响应闭环”意味着，当系统检测到一台属于国家重点航空航天研究机构高级研究员的工作站（高价值资产）

出现异常时：EDR 终端检测到一个伪装成系统进程的恶意软件，正与某个已知具有国家背景的 APT 组织的 C2 服务器进行加密通信，同时该研究员的账户正尝试访问其数月未曾接触的涉密项目数据。AI 引擎识别出这是典型的 APT 攻击模式后，会自主决策并执行一套组合拳式的缓解策略：首先，通过微隔离技术将该工作站的端口在交换机层面进行隔离，阻断横向移动；同时，将恶意 C2 域名推送至云端 SWG，全局阻断所有用户的访问；最后，通过 API 调用终端 EDR 方案，强制终止恶意进程，并自动创建包含所有上下文的工单给 SOAR 平台，以供安全分析师复核。

全面零信任的核心是“永不信任，始终验证”，而下一代 SASE 通过自主决策，让这个框架成为一个“活的”、自适应的现实[42]。每一次访问请求，都不再是简单地匹配一条静态规则，而是由系统基于其统一知识图谱提供的丰富实时上下文进行一次即时的、自主的风险评估与访问决策。例如，一个刚刚通过多因素认证的工程师在访问常规文档时可能畅通无阻，但当其设备风险评分因后台检测到异常进程而略微升高时，系统可能会在他试图访问核心代码库时，自主决策要求其进行一次额外的生物识别验证。这种动态、上下文感知的访问控制，使得“最低权限访问”原则能够被前所未有地动态、精准地执行。

这种架构还将管理员的角色从“规则配置者”转变为“业务意图声明者”。他们只需用接近自然语言的方式定义高级目标，例如：“确保公司对欧盟公民数据的处理完全符合 GDPR 法规要求。”网络原生智能定的智能分析引擎则会自主地将这一合规意图翻译并编排成一

系列具体的、跨越多厂商、多地域设备的安全与网络配置：它会自动发现并标记所有包含欧盟个人身份信息（PII）的数据库和云存储；生成 ZTNA 策略，确保只有位于欧盟境内且属于特定访问组的用户才能访问这些数据；创建 DLP 和防火墙规则，阻止任何被标记的 PII 数据传输至欧盟以外的地区，并持续监控配置漂移，以始终满足最初声明的合规意图。

6.3 构建可验证的安全智能体系

可验证的安全智能体系是融合网络原生智能架构，通过分层架构实现自动化威胁检测与响应，并具备可追溯、可解释、可审计、可靠性质的网络安全防护系统[43]。该定义以网络原生智能为核心支撑，强调通过智能化手段提升安全运营的自动化水平，同时通过可验证性相关机制保障体系的可信度与可控性。

从技术内涵来看，体系的核心特征体现在三个层面：其一，技术架构层面，依托图驱动智能编排框架，实现对海量安全数据的实时处理与深度挖掘，支持从被动防御向主动预测的转变；其二，能力特性层面，具备自学习、自适应与自优化能力，能够基于历史威胁数据与实时情报动态调整安全策略，例如通过机器学习模型持续优化检测规则以应对新型攻击手段；其三，安全性质层面，以可验证性为核心，涵盖可追溯、可解释、可审计等关键属性[44]。其中，可验证性指信息在传输与处理过程中可被验证来源合法性与完整性，例如通过公钥基础设施（Public Key Infrastructure，PKI）系统颁发数字证书时，可借

助证书链验证确保实体身份的合法性；可审计性作为可验证性的重要组成部分，要求对所有网络操作与数据访问行为进行全面记录与追溯，例如记录用户登录、数据查询等操作日志以便事后审查。

与传统安全体系相比，该体系的核心差异体现在两方面：一是动态调整能力的提升，传统安全体系多依赖人工配置的静态规则，难以应对快速演变的威胁环境，而可验证的安全智能体系通过 AI 驱动的自学习机制，实现安全策略的自动化优化与动态适配，例如基于攻防对抗数据实时更新检测模型；二是全流程可验证性的强化，传统安全机制虽具备部分审计功能，但缺乏对威胁检测、响应、处置全流程的系统性验证框架，该体系通过整合可追溯、可解释、可审计等性质，构建从数据采集到决策输出的完整可信链路[45]，例如通过操作日志的规范化管理与不可篡改设计（如基于区块链技术的存证方案）增强数据可信度，确保安全事件的可追溯与责任可认定。这种特性使得体系在金融、能源等关键领域的复杂环境中能够有效应对多维度、高持续性的安全威胁，提升整体防护效能。

七、结语

本白皮书以应对数字化时代安全挑战为核心，系统阐述网络原生智能的设计理念、技术架构与实践场景，深入解析感知-理解-决策-响应的闭环机制，并通过全网 DDoS 攻击检测与缓解、路由安全等场景验证架构可行性，为安全网络一体化提供智能技术支撑。

网络原生智能发展需由动态威胁防御需求与智能技术演进双向驱动。紫金山实验室联合产业伙伴在业界首次实现基于图驱动引擎的意图化安全编排框架，通过基于 YANG/NETCONF 的可编排安全能力按需投送理念，引领网络原生智能的技术革新。

我们期待通过本白皮书的探索，凝聚更多行业力量。诚邀全球产学研伙伴携手突破轻量化 AI 推理、零信任架构融合等关键技术，共建具备自适应防护、可验证决策与全域协同能力的下一代智能安全网络基础设施，护航数字经济高质量发展。

附录 A：术语与缩略语

中文名称	英文缩写	英文全拼
访问控制列表	ACL	Access Control List
人工智能	AI	Artificial Intelligence
智能运维	AIOps	Artificial Intelligence for IT Operations
高级持续性威胁	APT	Advanced Persistent Threat
自治系统	AS	Autonomous System
自治系统提供商授权	ASPA	Autonomous System Provider Authorization
边界网关协议	BGP	Border Gateway Protocol
BGP 监控协议	BMP	BGP Monitoring Protocol
指挥、控制与情报	C2I	Command, Control, and Intelligence
有向无环图	DAG	Directed Acyclic Graph
分布式拒绝服务攻击	DDoS	Distributed Denial of Service
域名生成算法	DGA	Domain Generation Algorithm
数据防泄露	DLP	Data Loss Prevention
数据处理单元	DPU	Data Processing Unit
端点检测与响应	EDR	Endpoint Detection and Response
通用数据保护条例	GDPR	General Data Protection Regulation
基于虚拟机监控器的安全	HBS	Hypervisor-Based Security
入侵防御系统	IPS	Intrusion Prevention System
物联网	IoT	Internet of Things
大语言模型	LLM	Large Language Model
管理信息库	MIB	Management Information Base
机器学习运维	MLOps	Machine Learning Operations
网络流	NetFlow	Network Flow

网络原生智能	NNI	Network-Native Intelligence
神经网络处理单元	NPU	Neural Processing Unit
开放神经网络交换	ONNX	Open Neural Network Exchange
ONNX 硬件集成接口	ONNXIFI	ONNX Interface for Integration
个人身份信息	PII	Personally Identifiable Information
服务质量	QoS	Quality of Service
融合以太网上的远程直接内存访问	RoCE	Remote Direct Memory Access over Converged Ethernet
资源公钥基础设施	RPKI	Resource Public Key Infrastructure
数据采集与监视控制系统	SCADA	Supervisory Control and Data Acquisition
安全访问服务边缘	SASE	Secure Access Service Edge
软件定义网络	SDN	Software-Defined Networking
软件定义广域网	SD-WAN	Software-Defined Wide Area Network
分段路由 IPv6	SRv6	Segment Routing IPv6
安全 Web 网关	SWG	Secure Web Gateway
Web 应用防火墙	WAF	Web Application Firewall
网络配置协议建模语言	YANG	Yet Another Next Generation
零信任网络访问	ZTNA	Zero Trust Network Access

参考文献

[1] 白宫科技政策办公室. 国家人工智能研发战略计划：2023 更新版[R]. 华盛顿: White House Office of Science and Technology Policy, 2023.

[2] 中共中央, 国务院. 关于构建更加完善的要素市场化配置体制机制的意见[EB/OL].(2020-04-09)[2024-12-01]. http://www.gov.cn/zhengce/2020-04/09/content_5500622.htm.

[3] 潘教峰, 万劲波. 构建现代化强国的十大新型基础设施[J]. 中国科学院院刊, 2020, 35(5): 545-554.国家发展改革委, 中央网信办, 工业和信息化部, 等. "东数西算"工程实施方案[EB/OL]. (2022-02-17)[2025-07-31]. http://www.gov.cn/zhengce/zhengceku/2022-02/17/content_5674198.htm.

[4] 国家发展改革委. 关于深入实施"东数西算"工程加快构建全国一体化算力网络体系的实施意见[EB/OL]. (2024-01-01)[2024-12-01]. https://www.gov.cn/zhengce/zhengceku/202401/content_6924596.htm.

[5] LI J, LIU L, ZHAO L, et al. Cyber security meets artificial intelligence: a survey[J]. Frontiers of Information Technology & Electronic Engineering, 2018, 19(12): 1462-1474.

[6] Wu J, Li R, An X, et al. Toward native artificial intelligence in 6G networks: System design, architectures, and paradigms[J]. arXiv

preprint arXiv:2103.02823, 2021.

[7] Banchs A, Fiore M, Garcia-Saavedra A, et al. Network intelligence in 6G: Challenges and opportunities[C]//Proceedings of the 16th ACM Workshop on Mobility in the Evolving Internet Architecture. 2021: 7-12.

[8] Wu W, Zhou C, Li M, et al. AI-native network slicing for 6G networks[J]. IEEE Wireless Communications, 2022, 29(1): 96-103.

[9] 华为技术有限公司. AI 原生 6G 网络的数据面设计[EB/OL]. [2024-12-01]. https://www.huawei.com/cn/huaweitech/future-technologies/data-plane-design-ai-native-6g-networks

[10] DAEMON Consortium. DAEMON: Network intelligence aDAptive sElf-Learning MObile Networks[EB/OL]. [2024-12-01]. https://h2020daemon.eu/.

[11] YANG Y, WU J, CHEN T, et al. Task-oriented 6G native-AI network architecture[J]. IEEE Network, 2023, 37(6): 272-279.

[12] Shi Y, Yang K, Jiang T, et al. Communication-efficient edge AI: Algorithms and systems[J]. IEEE communications surveys & tutorials, 2020, 22(4): 2167-2191.

[13] AGRAWAL A, KEDIA N, PANWAR A, et al. Taming Throughput-Latency Tradeoff in LLM Inference with Sarathi-Serve[C]//18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24). Santa Clara: USENIX Association, 2024:

1-18.

[14] SONG L, HU X, ZHANG G, et al. Networking systems of AI: On the convergence of computing and communications[J]. IEEE Internet of Things Journal, 2022, 9(14): 12520-12540.

[15] Song L, Hu X, Zhang G, et al. Networking systems of AI: On the convergence of computing and communications[J]. IEEE Internet of Things Journal, 2022, 9(20): 20352-20381.

[16] 傅懋钟, 胡海洋, 李忠金. 面向 GPU 集群的动态资源调度方法[J]. 计算机研究与发展, 2023, 60(6): 1308-1321. DOI: 10.7544/issn1000-1239.202220149

[17] MOKHTAR B. AI-enabled collaborative distributed computing in networked UAVs[J]. IEEE Access, 2024, 12: 89456-89470

[18] JOY M, VENKATARAMANAN S, AHMED M. AIOps in Action: Streamlining IT Operations Through Artificial Intelligence[J]. International Journal of Artificial Intelligence, 2024, 12(3): 45-62.

[19] BACCOUR E, MHAISEN N, ABDELLATIF A A, et al. Pervasive AI for IoT applications: A survey on resource-efficient distributed artificial intelligence[J]. IEEE Communications Surveys & Tutorials, 2022, 24(4): 2182-2204.

[20] IMT-2030(6G)推进组. 6G 网络原生 AI 技术需求白皮书[R]. 北京: IMT-2030(6G)推进组, 2022.

[21] JUNG B C. Toward artificial intelligence-native 6G services[J].

IEEE Vehicular Technology Magazine, 2024, 19(4): 18-25.

[22] CISCO Systems. What Is AIOps? Artificial Intelligence for IT Operations[EB/OL]. [2024-12-01]. https://www.cisco.com/site/us/en/learn/topics/artificial-intelligence/what-is-aiops.html.

[23] NGUYEN D C, DING M, PHAM Q V, et al. Federated learning meets blockchain in edge computing: Opportunities and challenges[J]. IEEE Internet of Things Journal, 2021, 8(16): 12806-12825.

[24] LIN X. Artificial intelligence in 3GPP 5G-advanced: A survey[J]. arXiv preprint arXiv:2305.05092, 2023.

[25] 商兴宇, 刘小欧, 杨明川. 人工智能原生网络发展趋势研究[J]. 信息通信技术与政策, 2023, 49(8): 1-8.

[26] NEPTUNE AI. How to optimize GPU usage during model training[EB/OL]. [2024-12-01]. https://neptune.ai/blog/optimizing-gpu-usage-during-model-training-with-neptune.

[27] ZHANG Y, MENG Q, HU C, et al. Revisiting congestion control for lossless ethernet[C]//21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24). Boston: USENIX Association, 2024: 1-18.

[28] PENG Y, WEI H, ZHONG X, et al. Barre: Empowering simplified and versatile programmable congestion control in high-speed AI clusters[C]//2025 USENIX Annual Technical Conference (ATC 25). Santa Clara: USENIX Association, 2025: 1-16.

[29] LIU S, WANG Q, ZHANG J, et al. NetReduce: RDMA-compatible in-network reduction for distributed DNN training acceleration[J]. arXiv preprint arXiv:2009.09736, 2020.

[30] ONNX Community. ONNX v1.18.0 Release Notes[EB/OL]. [2025-01-15]. https://github.com/onnx/onnx/releases/tag/v1.18.0.

[31] LE A. An MLIR-based Compiler for ONNX AI models[C]//2025 AsiaLLVM Developers' Meeting. 2025.

[32] JOSHUA C, KARKALA S, HOSSAIN S, et al. Cross-Platform Optimization of ONNX Models for Mobile and Edge Deployment[J/OL]. ResearchGate, 2025[2025-01-15]. https://www.researchgate.net/publication/392623112.

[33] Qualcomm Technologies Inc. Unlocking the power of Qualcomm QNN Execution Provider GPU backend for ONNX Runtime[EB/OL]. [2025-05-10]. https://www.qualcomm.com/developer/blog/2025/05/unlocking-power-of-qualcomm-qnn-execution-provider-gpu-backend-onnx-runtime.

[34] NEZAMI Z, HAFEEZ M, DJEMAME K, et al. Generative AI on the edge: Architecture and performance evaluation[J]. arXiv preprint arXiv:2411.17712, 2024.

[35] NAAYINI P. Building ai-driven cloud-native applications with kubernetes and containerization[J]. International Journal of Scientific Advances (IJSCIA), 2025, 6(2): 328-340.

[36] IPC. The Top 5 SD-WAN Trends and Advancements for 2025[EB/OL]. [2025-01-15]. https://www.ipctech.com/sd-wan-trends-for-2025/.

[37] SAXENA N, YADAV A R, TALWANDI N S. Beyond Intent: A Unified AI Framework for Self-Optimizing, Self-Securing, and Self-Healing Networks Using Generative AI, Federated Learning, and Neuromorphic Computing[J]. IJSAT-International Journal on Semantic Web and Information Systems, 2025.

[38] DAVID S. AI-Driven Network Management Systems: A Review of Intelligent Monitoring, Predictive Maintenance, and Self-Healing Capabilities[EB/OL]. ResearchGate, 2025[2025-01-15]. https://www.researchgate.net/publication/392666845.

[39] SHAJARIAN S, KHORSANDROO S, ABDELSALAM M. A Survey on Self-Running Networks: Concepts, Components, Opportunities, and Challenges[J]. Authorea Preprints, 2024.

[40] Zscaler Inc. 5 Predictions for Zero Trust and SASE in 2025[EB/OL]. [2025-01-15].https://www.zscaler.com/blogs/product-insights/5-predictions-zero-trust-and-sase-2025-what-s-next.

[41] FOPA MAMENE M. Secure Access Service Edge (SASE): Architecture, Implementation, and Performance Evaluation[D]. 2024.

[42] AJISH D. The significance of artificial intelligence in zero trust technologies: a comprehensive review[J]. Journal of Engineering

Science and Innovative Technology, 2024.

[43] NWEJE U. Blockchain Technology for Secure Data Integrity and Transparent Audit Trails in Cybersecurity[J]. International Journal of Research and Publication Reviews, 2024.

[44] QADER K S, CEK K. Influence of blockchain and artificial intelligence on audit quality: Evidence from Turkey[J]. Heliyon, 2024, 10(10): e30166.

[45] BESHARAT P. AI and Blockchain, Enhancing Security, Transparency, and Integrity[EB/OL]. ResearchGate, 2024[2025-01-15]. https://www.researchgate.net/publication/392626580.