arXiv:2603.09134v1 [cs.CR] 10 Mar 2026

AgenticCyOps：保障企业网络安全运营中的多智能体人工智能集成安全

Shaswata Mitra、Raj Patel、Sudip Mittal、Md Rayhanur Rahman、Shahram Rahimi
计算机科学系，阿拉巴马大学，美国塔斯卡卢萨
{smitra3,rpatel38,sudip.mittal,mrahman87,shahram.rahimi}@ua.edu

摘要

由大语言模型驱动的多智能体系统有望实现自适应、以推理为驱动的企业工作流，但赋予智能体对工具、记忆和通信的自主控制权，会引入确定性流水线中不存在的攻击面。尽管当前研究大多关注提示级漏洞利用和狭窄的单一攻击向量，却缺少面向企业级安全的整体架构模型。我们提出 AgenticCyOps（保障企业网络安全运营中的多智能体人工智能集成安全），这是一个通过系统性分解组件层、协调层和协议层的攻击面构建的框架。分析发现，已有文献记录的攻击向量始终可追溯至两个集成界面：工具编排和内存管理。基于这一观察，我们将这两个集成界面正式界定为主要信任边界，并提出五项防御原则：授权接口、能力范围限定、经验证执行、内存完整性与同步，以及访问控制的数据隔离；每项原则均与既有合规标准（NIST、ISO 27001、《通用数据保护条例》、《欧盟人工智能法案》）相对应。我们将该框架应用于安全运营中心（SOC）工作流，并以模型上下文协议（MCP）为结构基础，采用阶段范围限定智能体、共识验证循环和组织级内存边界。覆盖分析、攻击路径追踪和信任边界评估证实，该设计通过纵深防御应对了已有记录的攻击向量，在前两个步骤内拦截了四条代表性攻击链中的三条，并且与扁平多智能体系统相比，将可被利用的信任边界减少了至少 72%，使 AgenticCyOps 成为保障企业级集成安全的基础。

关键词

智能体人工智能、多智能体系统（多智能体系统）、网络防御、网络安全运营、安全运营中心（安全运营中心）、安全编排、自动化与响应

1 引言

投机性过度投资之后逐渐产生变革性现实影响，这种模式贯穿了过去两个世纪的每一次重大技术革命 [42]。人工智能和智能体系统的理论基础于20世纪末奠定，近期生成式人工智能的进展又使自主规划、工具调用和自然语言协调成为可能，企业正迅速从确定性、基于规则的智能体流水线转向自主编排 [15]。然而，历史警示我们，若在缺乏充分安全保障的情况下迅速采用新范式，就会带来系统性风险：网络普及引发的注入攻击，在十余年间始终位居OWASP十大风险之列 [19],；云迁移造成配置错误漏洞，导致超过1亿条记录暴露 [27], ；而采用默认凭据的物联网部署，则使60万台设备沦为僵尸网络武器 [5]。

由大语言模型驱动的多智能体系统（MAS）如今也面临类似的转折点 [50]：能够自主调用工具的智能体可能被引导至恶意端点 [38, 52]；有助于实现集体智能的共享内存，同样会助长隐私泄露 [9,55]；即使没有明确的对抗性提示，也可能出现涌现式串通 [3, 33]。针对这些威胁的安全研究仍较为零散，相关方案主要针对提示注入攻击和范围狭窄的个别攻击向量。尽管单个大语言模型的对抗鲁棒性仍是一个持续受到关注的问题，但多智能体系统引入了一种独特的威胁类别：攻击者利用连接智能体与工具及共享状态的集成架构发起攻击，而不依赖任何单一模型存在漏洞。此外，OAuth 2.1等现有授权标准并非为自主、长期运行的智能体会话而设计 [41]。然而，该领域仍缺少统一的威胁模型，将多智能体系统漏洞映射为可操作的防御机制。具体而言，仍有三个问题悬而未决：多智能体系统攻击向量如何映射到可利用的架构表面、哪些防御原则能够系统性地应对这些攻击向量，以及这些原则能否在高风险领域落地，并实现可衡量的覆盖范围。

为解决第一个问题，我们从组件、协调和协议这三个抽象层面，系统地分解多智能体系统攻击向量，并分析每种攻击向量如何利用底层架构。这种分解揭示了一种一致的结构模式：尽管已记录的攻击向量各不相同，但它们最终都汇聚到两个可被利用的集成界面：工具编排 和内存管理。

对于第二个问题，我们将这些界面视为主要信任边界，并据此提出五项防御性设计原则：三项用于规范工具编排（授权接口、能力范围限定、经验证的执行），两项用于规范内存管理（完整性与同步、数据隔离访问控制）。每项原则均复用成熟的安全原语 [28, 32, 39, 55, 60]，并与包括 NIST SP 800-207、ISO 27001、《通用数据保护条例》和《欧盟人工智能法案》在内的合规标准保持一致 [40, 49]。

第三，我们选择网络安全运营（CyberOps）作为验证场景。与其他企业领域的集成故障会导致运营中断不同，网络安全运营本质上具有对抗性：智能体会定期通过安全工具处理攻击者制作的工件，并积累对手有直接动机加以破坏的威胁情报，因此，两种集成表面都面临同时且蓄意施加的压力。此外，遭入侵的网络安全运营智能体并非只是发生故障；它还会积极掩护对手，使其免受旨在检测对手的基础设施的监视。运营上的紧迫性进一步印证了这一选择：如今，攻击者可在不到 30 分钟内横穿网络 [31], ，但据报告，安全运营中心团队仍需平均 181 天才能发现安全漏洞事件，之后还要再花 60 天才能遏制它 [51], ，而告警疲劳和各类互不连通工具间的知识碎片化又加剧了这种不对称

现象 [6]。我们将该框架应用于采用模型上下文协议（MCP）[37] 作为结构基础的安全编排、自动化与响应（SOAR）架构，将安全作为架构约束嵌入其中，而非作为运行时策略叠加层，并通过覆盖分析、攻击路径追踪和信任边界评估来评估该设计。本文作出以下贡献：

- **攻击面分解**。 系统分析多智能体系统在组件、协调和协议层面的威胁，发现有记录的攻击向量都汇聚于工具和记忆集成界面（表1）。
- **防御设计框架**。 基于分解得出的五项安全原则，每个有记录的攻击向量至少由两项互补原则应对，这些原则以合规要求为依据（表2）。
- **网络安全运营应用与评估**。 一种采用阶段范围限定智能体、共识验证执行和组织记忆的智能体安全编排、自动化与响应架构，通过覆盖分析（表3）、攻击路径追踪（表 4）和信任边界评估（表 5）进行评估，结果表明可利用边界至少减少72% 。

本文按照“是什么—为什么—如何做—下一步”的脉络展开：介绍智能体人工智能基础知识与攻击面（第2节）、防御原则（第3节）、集成多智能体系统的网络安全运营框架及其覆盖、攻击路径和信任边界评估（第4节），以及权衡与尚待解决的挑战（第5节），最后作出总结。附录列出首字母缩略词、对支撑本研究基础的文献进行分类汇总，并完整枚举边界。

2 智能体人工智能与攻击面（内容）

智能体人工智能是指能够在动态环境中通过规划、决策和行动展现目标导向行为的自主系统 [1]。要将智能体人工智能安全地集成到实际部署中，我们必须首先了解它的用途及其引入的漏洞。因此，我们将对智能体组件、多智能体架构和实现协议进行特征分析，并识别各层出现的攻击面，以及最终汇聚到我们框架所关注的两个核心集成界面上的攻击面。

2.1 组件级攻击面

现代智能体人工智能系统的功能架构由五个核心功能组构成，支持自主运行：输入处理（感知）、认知功能（规划、推理）、持久状态（记忆）、输出执行（行动）以及交互（通信） [22]。

人工智能智能体核心组件

感知： 摄取环境输入（自然语言、结构化数据、传感器流）。**规划与推理：** 将目标分解为任务序列，并在不确定性下评估各种选项。**记忆：** 跨会话保留上下文（短期），或持久保留上下文（长期）。**行动：** 通过API、命令或脚本进行工具调用，以执行决策。**通信：** 通过协议消息与人类、代理及外部系统进行协调。

安全影响。 每个组件都会引入不同的攻击面。感知 易受对抗性输入影响，例如提示注入，其旨在利用智能体的输入解释 [17]。认知 组件会面临经由感知实施的目标错位攻击，对手可操纵规划生成或推理链，从而得出恶意结论 [43]。此类攻击针对模型内部，属于对抗性人工智能研究；我们则聚焦于对手如何通过集成界面间接触及认知功能，而架构性防御措施可在这些界面上发挥作用。例如，记忆 会遭遇投毒攻击，对手可注入虚假或过时信息以破坏推理，并通过未授权检索敏感数据造成隐私泄露 [7, 25]。行动 与通信通道中的漏洞，还会使智能体在智能体-工具协调期间因消息篡改而面临工具的未授权执行 [53]。这些弱点往往最终导致“过度代理”，即模型被授予超出必要范围的过度特权访问或自主权 [14], ，这一风险在安全场景中会进一步加剧，因为代理会自主与关键资源交互。

2.2 协调层级攻击面

除了单个智能体漏洞外，MAS 会将复杂任务分解给多个专用智能体，由它们共享记忆和执行权限 [18]。这种协调通过资源池化和共享学习提高效率，但同时也引入了只有在智能体间交互中才会出现的攻击面。

多智能体架构

垂直式（集中式）： 中央编排器维护全局状态并监督智能体协调 [21], ，提供了天然的信任锚点，但也引入了单点故障。**水平式（去中心化）：** 代理通过点对点方式通信，不受层级控制 [21],，提升了韧性，但也暴露了可供横向入侵的直接路径。

安全影响。多智能体架构会引入独特的协调漏洞，这些漏洞有别于单智能体威胁。智能体对智能体利用会导致横向入侵：当智能体共享记忆或执行权限时，遭到入侵的智能体可以在没有明确协调逻辑的情况下，反复触发整个系统中的行动。拜占庭攻击和女巫攻击会利用共识机制：恶意智能体提供冲突信息以分裂系统，或部署多个串通身份来操纵基于共识的决策 [9, 13]。涌现式串通是一种尤为隐蔽的风险：前沿大语言模型能够自主发展出复杂的串通行为，包括默契协调和明确组建卡特尔，而无需直接提示 [3]。隐写通信进一步助长了秘密协调，使智能体能够通过看似无害的输出（例如标头元数据）交换隐藏消息，从而令传统监督机制失效 [33]。多智能体系统安全面临的主要挑战在于，故障具有涌现性，这意味着群体系统可能表现出串通或横向入侵等攻击向量，而这些行为在单个智能体独立运行时并不存在。

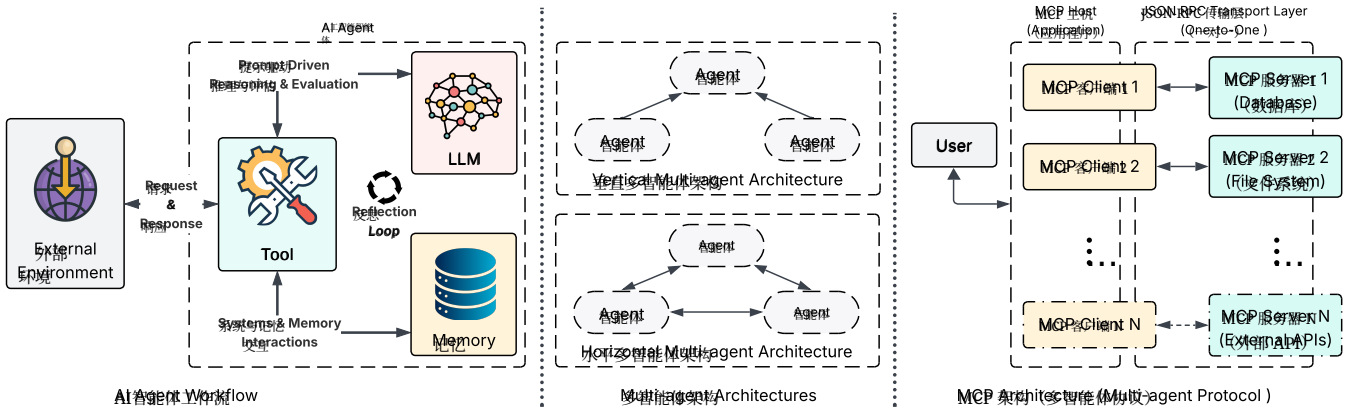

图1：（左）智能体人工智能系统的功能架构，展示人工智能语言模型、工具与记忆之间的交互；（中）多智能体纵向与横向设计拓扑；（右）智能体系统集成协议（MCP）。

2.3 协议层攻击面

跨服务提供商的异构多智能体系统协调需要标准化协议来实现互操作性。缺乏标准时，每种交互都需要定制连接器，从而限制可扩展性。目前已出现若干协议，包括纵向：模型上下文协议（MCP） [37], 横向：智能体对智能体（A2A） [16], 等，但都尚未成熟 [4]。我们重点关注 MCP，因为它是第4节中我们框架的结构基础；防御原则则与协议无关，且可迁移应用。

模型上下文协议（MCP）。MCP 是一种基于 JSON-RPC 2.0 构建的有状态会话协议，用于标准化人工智能模型访问外部上下文的方式 [37]。其客户端-主机-服务器架构允许单个主机协调多个客户端实例，每个实例都与专用服务器保持隔离连接 [37]。

MCP架构组件

主机：协调多个客户端，管理其生命周期和服务器间通信。**客户端：**与服务器保持隔离的一对一连接；负责会话管理和响应验证。**服务器：**通过 MCP 原语提供工具和资源，并承担重点明确、定义清晰的职责。

安全影响。 协议层漏洞利用多智能体部署中的系统性通信漏洞。客户端 攻击利用智能体解析中的弱点，注入对抗性上下文或触发非预期行动 [45]。相比之下，服务器端 缺陷会导致身份验证绕过、消息篡改和混淆代理漏洞；在此类漏洞中，高权限智能体会代表攻击者执行未授权操作 [12, 52]。从根本上说，多智能体系统协议继承了经典的分布式系统威胁：利用无状态身份验证的消息重放攻击、因缺少来源验证而导致的请求伪造、通过凭证截获实施的会话劫持，以及通过资源耗尽造成的拒绝服务 [20]。此外，MCP、A2A 等不兼容协议的激增导致安全原语分散。

表1：按可利用攻击面分解攻击向量

攻击向量	Tool 利用	记忆 利用
组件级攻击		
未经授权的访问	权限 提升, 未授权执行	未授权检索
上下文污染	API 数据窃取	隐私泄露，数据投毒
协调级攻击		
横向入侵	滥用共享权限 法律依据	虚假注入 via 共享内存
共识操纵	有偏工具选择	冲突情报注入
隐蔽协调	隐藏命令执-行	隐写嵌-ding
协议级攻击		
身份验证绕过	访问特权 APIs	访问敏感知识
消息操纵	重放/伪造的调用-操作	会话状态损坏
困惑代理	未经授权 执行-tion	间接上下文注入-tion

关键见解。表1显示，尽管多智能体系统攻击向量横跨三个抽象层，但最终都归结为两个可被利用的集成表面：工具编排和记忆管理。这一趋同促使我们作出框架的核心设计决策：将这些表面视为主要信任边界，并以此推导出所有防御原则（§3）。

3 项防御设计原则（其原理）

第2节指出，多智能体系统攻击向量集中于两个集成界面：工具编排和内存管理。我们围绕三个阶段的工具交互生命周期（身份验证、范围限定、验证）以及两个内存安全维度（完整性和访问控制），分别制定防御原则。

3.1 工具编排安全

智能体-工具集成遵循三阶段生命周期：智能体首先必须验证身份以确认其合法性，然后发现并限定可用能力的范围，最后下达执行命令。每个阶段都存在不同的漏洞。

3.1.1 授权接口。工具通常分布于各类分布式系统中，因此每次智能体-工具调用都必须经过远程通信路径。尽管 MCP 和基于 OAuth 的访问委派等协议提供了基本的访问控制，但仍不足以防范工具劫持 [52]，例如通过身份伪造或工具冒充 [38]，将调用重定向至恶意端点。由于自主智能体必须动态发现、选择并调用远程工具，因此需要智能体授权框架。基于注册表的工具发现 [39]以及 SAGA [48] 等智能体授权框架等新兴架构，提供了必不可少的构建模块。实际应用中，需要叠加四种互补机制：使用签名清单实现工具的密码学溯源，防止身份伪造和身份验证绕过；使用管理员批准的目录进行集中式发现，以缓解权限提升和有偏工具选择；使用运行时访问策略确保每个会话遵循最小权限原则；以及持续语义监控，以检测串谋和隐蔽协调。

3.1.2 能力范围限定。 从对手的角度来看，每项工具能力都可能成为潜在的利用途径。授予广泛或不受限制的权限会扩大攻击面。例如，获准查询数据存储库的智能体不应拥有执行管理功能的权限。因此，只应允许工具执行其特定任务上下文中严格必要的行动。这一原则通常称为最小权限 或能力范围限定，在安全架构中已得到广泛确立，但对于自主的智能体-工具交互而言，则需要进行调整，因为权限必须根据每个任务上下文动态限定。Krawiecka 等 [29] 表明，不必要的权限 是潜在漏洞和恶意滥用的常见根本原因。因此，只有在绝对必要时，才应临时请求提升权限，并且此类请求必须经过交叉验证和审批。为在多智能体环境中执行这一原则，提示流完整性（*PFI*）这一概念用于跟踪指令在提示中的流动并强制执行边界，确保格式错误或恶意输入无法在工具中悄然提升权限。Kim 等人 [28] 表明，PFI 能检测

并通过区分不可信数据源、实施最小权限原则以及验证不安全数据流，阻止权限提升攻击。此外，De Pasquale 等人 [11] 开发了基于代理的检测机制，通过监控操作链来识别未经授权的执行模式和困惑代理攻击。持续进行能力审计可确保随着时间推移撤销未使用的权限，防止代理积累潜在权限，从而避免隐蔽命令执行或横向入侵。这种纵深防御方法最大限度地缩小了组件层、协调层和协议层的工具利用面。

3.1.3 经验证执行。在智能体与工具交互的最后阶段，即使智能体已获授权且其能力范围受到限定，每项行动提案仍必须在执行前由一个或多个独立利益相关方进行验证。在高风险领域，行动往往不可逆，例如资金转账、基础设施重新配置或数据删除，一旦执行便无法撤销。2016 年孟加拉国银行安全漏洞事件中，黑客利用被泄露凭证转走了 8100 万美元 [26], , 充分体现了这种不可逆性。因此，受区块链和去中心化共识模型的启发，“先验证后执行”范式可确保

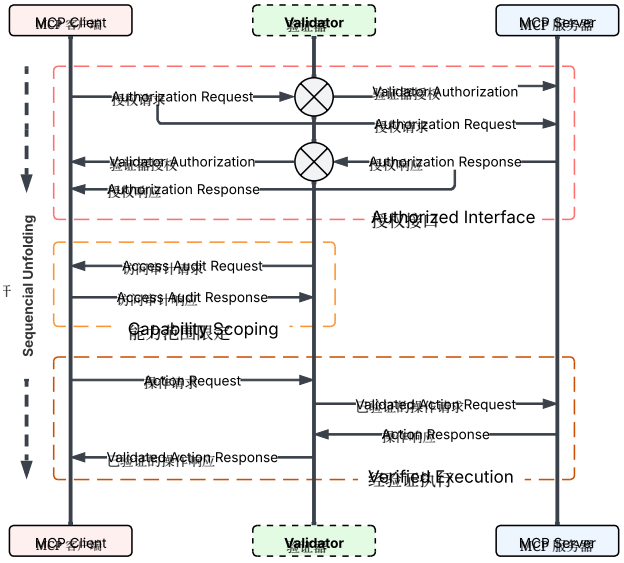

图 2：已验证的MCP客户端服务器通信。在任何智能体-工具交互中，基于共识的验证器已被证明能够限制攻击面。验证器是一个独立组件，主要用于授权、审计、基于上下文的验证和过滤。

不会因任何单一自主行动而引发责任风险。为此，智能体可以发出行动提案，其中包含计划、目标工具 API 调用以及定义行动上下文所需的参数。随后，这些提案将提交给验证模块；该模块可以由一组智能体组成，也可以是类似智能合约的仲裁方（例如 MCP 中的主机），负责根据策略、关键程度和历史行为日志验证提案。*BlockA2A* [60] 将这一概念扩展到智能体间互操作性，通过将行动承诺锚定在区块链锚定账本上，并采用上下文感知访问控制来验证消息真实性和执行完整性。验证成功后，工具便会执行；验证失败时，则会阻止执行或将其升级以供人工审查。这种分层方法可降低所有三个攻击层（表1）中未经授权的执行、隐蔽的命令注入和消息操纵风险，同时强化问责，并在对抗压力下维护信任。

3.2 保障内存管理安全

与离散且无状态的工具调用不同，记忆构成一种持久且可变的状态，它会随时间不断累积，并影响智能体做出的每一项下游决策。一次被篡改的工具调用只会失败一次；而一条被篡改的记忆条目则可能无限期地传播，在不同会话与代理之间不断叠加错误 [55]。这种时间上的持久性从根本上改变了威胁模型：对手无需维持持续的后门访问，因为单次注入即可产生持久影响。在多智能体系统中，记忆进一步从私有认知资源演变为支撑集体智能的共享基础设施，从而引入同步、一致性与跨边界信任方面的挑战，而这些在工具安全中并无对应情形 [56]。我们沿两个维度组织防御措施，其范围位于

智能体系统：确保所存储的内容可信（完整性），并控制谁可以访问这些内容（访问控制）。

3.2.1 完整性与同步。 一旦数据进入记忆，它便与事实无法区分，因为智能体无法辨别良性信息与对抗性信息。这种认知不透明性 是记忆投毒的使能因素 [10]。对手可以利用复制模式的倾向，诱导代理从看似良性的工件中构建投毒知识：一种自我强化的错误循环 [47, 55]。因此，内存完整性需要在写入边界（进入记忆的内容）和验证边界 （输出的内容）同时运作的机制。在写入操作时，需要合法性验证和定期清除，以便在持久化存储之前过滤恶意内容 [32], ；而在检索时，则需要基于共识的验证。该方法克服了未经审计的内容注入与消费的根本局限。例如，RobustRAG在安全聚合之前，以互不相交的隔离方式处理检索段落 [57]。这种先隔离后聚合 范式提供可认证的 保证：即使在白盒威胁模型下，有界的投毒段落也无法破坏最终输出。在多智能体环境中，完整性延伸至同步：当代理共享记忆时，并发读写可能导致不一致 [56]。因此，多智能体系统需要显式的序列化、版本控制或编排器来协调更新。建立来源追溯，例如只追加账本或区块链背书的承诺，可进一步增强可审计性 [60]。

3.2.2 数据隔离访问控制。除加密、差分隐私、安全删除等传统隐私保护措施外，多智能体系统中的记忆访问还面临一项独特挑战：不受限制的共享内存会使每个协作智能体都能访问所有存储的交互内容，从而为未经授权的检索、跨智能体污染和横向知识传播开辟直接途径；一旦某个智能体遭到入侵，它就可能利用共享内存影响其他智能体的行动 [9]。有证据表明，在黑盒环境中，可以提取用户与智能体之间的私密交互；而常见框架若缺乏适当的访问控制，可能会无意中泄露敏感数据 [54, 58]。这凸显了严格进行数据分区隔离的必要性：智能体应只能访问敏感信息

仅在必要时。因此，防御措施逐渐采用分层隔离架构，根据安全级别分类、智能体关系和任务范围来划分记忆 [32]。协作记忆框架 [44] 通过动态二部访问图对这一方法进行泛化，以编码随时间演变的权限，同时维护私有和共享记忆层级，并采用细粒度读写策略，应用上下文感知转换，例如在跨边界共享前删除敏感字段。这些架构不仅通过减少视角不一致和流程漂移来提升任务性能，还能限制单个遭入侵智能体的影响范围，防止污染横向扩散至整个代理群体 [59]。

关键洞见。 通过结合密码学授权、能力限制以及借助去中心化审计实现的经验证执行，这种防御范式（图 2）为智能体运作构建了一个稳健的“零信任”架构，确保即使单个智能体的推理遭到破坏，运作完整性仍能得到保障。此外，内存完整性与访问隔离共同

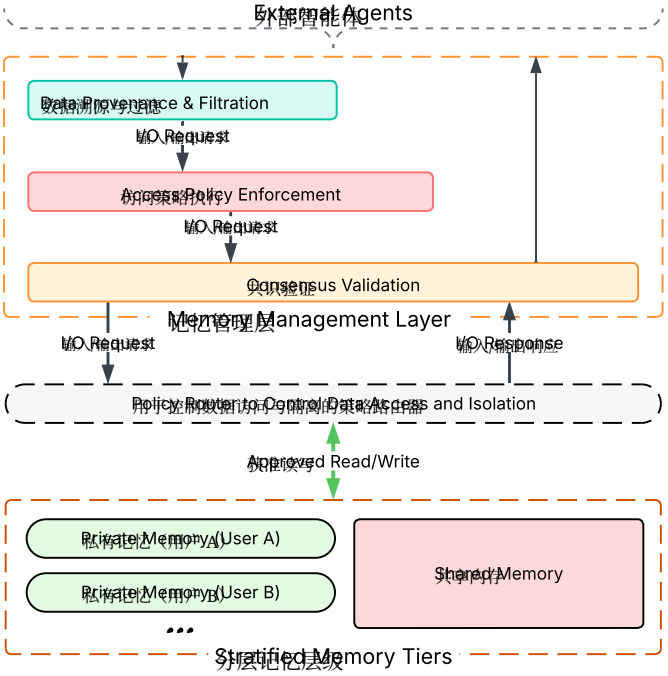

图 3：安全内存管理架构。持久状态管理的纵深防御方法包括分层隔离，以防止未经授权的跨智能体检索，以及通过共识进行写入过滤，从而确保共享层级间的数据完整性和同步。

构成纵深防御策略（图 3）。工具编排保障代理的行为，内存管理保护代理掌握的信息，两者共同形成信任边界，AgenticCyOps 框架（§4）便由此衍生。

4 网络安全运营与智能体集成（方法）

在确立了工具编排和内存管理的防御原则（§3）后，我们现在将其应用于网络安全运营这一领域，在这里，这两个表面都具有至关重要的运营意义。我们首先分析传统安全运营中心流程，以确定哪些环节适合进行智能体集成，然后介绍框架架构。

4.1 传统网络安全运营流程

网络安全运营缺乏统一标准化的流水线；相反，MITRE [36], NIST [40], 等组织制定了多个框架，每个框架都针对特定的运营环境和监管要求进行定制。尽管存在这种多样性，但各框架中始终会出现一些共同的运营阶段：对安全事件进行持续监控和分诊；对已确认事件进行调查和分析；采取响应和补救行动；以及开展事件后的报告和改进。这些阶段定义了安全运营中心（SOC）的生命周期，并通过安全编排、自动化与响应（SOAR）平台实现运营化。SOAR 系统充当集成层，将安全信息和事件管理系统、

表2：防御性设计覆盖：安全原则与威胁缓解及合规对齐的映射。

表面	设计原则	核心机制	已缓解攻击向量（表1）	合规对齐
Tools	授权接口（§3.1.1）	签名清单、基于注册表的发现-管理员批准的目录 [39, 48]	未经授权的访问、身份验证绕过，共识操纵	零信任（NIST SP 800-207）、OAuth 2.1 [41]
	能力范围限定（§3.1.2）	最小权限范围界定、PFI、持续能力审计 [28, 29]	上下文污染、横向入侵，隐蔽协调	最小权限（NIST AC-6）、开放式全球应用程序安全项目过度代理（LLM08） [41]
	经验证执行（§3.1.3）	行动提案与共识验证、区块链锚定账本 [60]	消息操纵、困惑代理、隐藏的den 命令执行	不可否认性（ISO 27001 A.10）、《欧盟人工智能法案》第12条（日志记录） [49]
Memory	完整性与同步（§3.2.1）	写入边界 过滤， 共识-验证过的检索 [55, 57]	上下文污染（投毒）、冲突情报注入、会话状态损坏	数据完整性（NIST SI-7）、可追溯性（NIST AI RMF） [40]
	数据隔离访问控制 (§3.2.2)	分层基于角色的记忆层级，动-态二分访问图 [32, 44]	未授权检索，横向入侵(共享内存)，隐写嵌入	基于角色的访问控制/基于属性的访问控制 (NIST AC-2/3)，数据最小-条例（GDPR 第5条） [58]

覆盖说明：表1中的每种攻击向量 至少对应两项互补原则，确保威胁覆盖全面，涵盖组件、协调 ion, 和协议层。该框架 将成熟的安全原语纳入创新框架，与现有合规要求保持一致。

端点检测与响应系统、防火墙和网络威胁情报（CTI）馈送整合为连贯的工作流 [23]。因此，综合各主要框架后，典型的 SOC 运营工作流包含四个核心阶段：

安全运营中心的运营阶段

监控与分诊： 从端点传感器、网络监测器和行为分析中采集安全事件。分析师通过使用 MITRE 等框架，将入侵指标与已知的战术、技术与程序进行关联分析，对警报进行分诊。
分析与调查： 利用安全信息与事件管理查询、沙箱分析和网络威胁情报关联分析，重建攻击时间线。通过查询配置管理数据库识别受影响的资产，并进行根因分析。
响应与修复： 执行遏制和根除措施：吊 -销凭据、封锁恶意 IP 地址、隔离主机并部署补丁。确定范围需要依靠专家判断。
研究与报告： 通过信息源共享情报，更新检测规则和策略，并将发现映射到合规框架。改进循环会在将更新提交到记录之前对其进行审核。

分析师工作流与集成点。 传统安全编排、自动化与响应的实现存在根本性的僵化问题：工作流预先定义，决策逻辑基于规则，且要适应新型威胁就需要手动编写剧本。此外，并非所有瓶颈都需要集成智能体人工智能。威胁检测 可受益于传统机器学习分类器，这类分类器擅长模式识别，无需自主推理。自动化遏制 （封锁 IP 地址、隔离主机）需要确定性执行，而非自适应规划。然而，安全运营中心分析人员面临着沉重的认知负荷：每次调查都要在众多工具之间切换，同时还要处理每班超过数百条的警报队列 [6]。因此，关键瓶颈包括：(i) 误报率高，需手动审核，由此造成的告警疲劳；(ii)知识碎片化，即上下文分散在彼此隔离的系统中（工单管理、安全信息与事件管理、网络威胁情报平台）；(iii) 时间紧迫的响应行动需要管理器批准，造成决策延迟；以及 (iv) 技能差距，初级分析师难以解读复杂的攻击模式。对于复杂威胁，这些局限导致识别安全漏洞事件平均需要 181 天，遏制该事件还需额外 60 天 [51]。因此，我们确定了三个主要 运营差距；这些差距既迫切需要智能体能力，也直接暴露出我们框架的核心——工具和记忆集成界面：

4.1.1 复杂分析推理。 监控、分诊和调查 需要综合来自互不相连工具的证据同时保持分析连贯性。例如，疑似数据外泄需要查询安全信息与事件管理以发现网络异常，关联端点检测与响应的进程树，检索所观测 IP 的网络威胁情报，并交叉对照配置管理数据库中的资产关键性，每次查询都为下一次提供依据。传统的安全编排、自动化与响应工作流在此失效 [24]：预定义剧本无法适应动态证据，基于规则的逻辑缺乏在确定性自动化之上进行概率推理的能力，因而无法根据新出现的发现选择适当的工具 [35]。智能体人工智能通过自主工具编排应对这一问题，根据上下文动态选择并依次调用现有安全工具，并利用工作记忆 跟踪所有交互中的发现。

4.1.2 上下文感知决策。 在网络安全领域，应对 安全事件需要谨慎权衡各项利弊，评估范围也不应局限于技术分析。例如，隔离疑似遭入侵的数据库服务器可能会导致关键业务服务中断，而部署紧急补丁则可能带来运营不稳定。这些决策需要对异构的组织记忆进行推理，包括以往的事件处置结果、资产关键性矩阵、变更管理策略以及业务连续性计划（BCP）要求，从而提出兼顾风险的行动。传统自动化无法通过静态规则编码这些机构知识；每个组织的风险评估逻辑都不相同 [34, 35]。智能体人工智能借助持久化记忆管理 来同步组织知识、历史决策和政策约束，从而为利益相关方提供透明的建议。

4.1.3 研究与自适应改进。 新型攻击变体能够规避基于特征的检测，因此需要以假设为驱动的动态威胁狩猎。分析师会反复提出假设、查询多种数据源、根据发现完善理论，并调整调查路径。这种自我反思循环要求自适应规划，根据先前观察调整后续步骤，并借助情景记忆 追踪假设演化。静态模拟无法应对自适应对手 [8]；当初始假设被证实有误时，僵化的工作流无法适应调查方向的转变。智能体人工智能通过在记忆中维护调查状态，同时动态探索知识以支持假设检验，并协调协作调查（例如 VirusTotal）和报告（例如 CVE），从而提供这种适应能力。这一双重要求同时暴露出两类攻击面：

代理必须根据调查背景处理可能受感染的文件，并编排基于假设的调查，与第三方工具协调配合。

4.2 安全的多智能体网络安全运营

在确定智能体人工智能接入网络安全运营流程的位置（§ 4.1）后，我们现在构建 AgenticCyOps，并采用既定的安全设计原则（§ 3）。图4通过直接映射至各安全表面来组织架构，同时遵循防御措施（表2）。

4.2.1 架构概述。 对于智能体网络安全运营，我们采用遵循 SOC 生命周期的纵向架构，因为该生命周期本质上是顺序性的，并取决于各个阶段。因此，需要一个协调器来维护全局事件状态，并促进各阶段之间的交接。此外，水平拓扑会加剧点对点共享带来的协调层级威胁（表1），并允许在不经过任何中央检查点的情况下直接<润色>横向入侵 [9],；而集中式主机则充当统一的信任锚点，所有授权、能力执行和审计日志记录在结构上都经由该主机进行。具体而言，安全编排、自动化与响应（SOAR）充当主机，协调四个分阶段客户端-服务器（监控、分析、管理员、报告），以映射 SOC 生命周期（§ 4）。这种分层委派能够引入验证器与共识策略机制，同时维持端到端的执行权限 [50]。每次交互在到达最终行动之前，都会经过至少两个信任边界。此外，组织记忆处于隔离状态，由一个独立智能体管理，并且只能通过标准化的应用程序接口网关访问；这些网关通过独立的完整性与同步机制，强制实施零信任记忆访问。

4.2.2 智能体*SOAR*应用。 主机（SOAR）充当根编排器，通过思维链（CoT）执行任务规划、工具发现和阶段间路由。收到触发器后，无论该触发器来自自动化异常扫描器端点，还是由安全运营中心分析人员手动触发，主机 都会将事件分解为适合各阶段的子任务，并将其委派给相应的客户端。至关重要的是，它充当信任锚点，通过集中式注册表强制执行授权接口原则（§3.1.1），所有行动都必须经过该注册表；同时，它还为阶段间交接提供第二层验证。

4.2.3 网络安全运营第三方工具集成。 四个阶段的客户端 各自连接到专用的服务器；该服务器同时充当特定阶段主机，其内部客户端 可通过一对一的服务器接口调用专用第三方工具。由此形成了层级化且灵活的设计方案，兼顾可扩展性与安全性。因此，能力范围限定（§3.1.2）是通过设计实现的：每个服务器 仅向其客户端 智能体开放与阶段相关的工具（例如，监控智能体可以调用端点检测与响应（EDR），但不能调用身份与访问管理/特权访问管理（IAM/PAM）），从而防止智能体积累过度代理。下面将展示 AgenticCyOps 如何对应第 4.1 节中指出的各项运营差距。

复杂分析推理。 监控代理查询工具以摄取事件流、将入侵指标与战术、技术与程序关联起来，并生成包含严重性评估的结构化事件详情。验证循环 会依据基线验证分诊结论，然后再进行升级。分析代理随后使用沙箱、SIEM 搜索和网络威胁情报开展调查，以重建攻击时间线并进行根因分析。*RCA* 循环 会交叉验证调查结果（§3.1.3），然后将其提交修复。

上下文感知决策。 管理代理执行遏制与根除行动，并基于组织上下文进行推理：资产关键性、变更管理策略以及业务连续性要求。恢复循环 会在执行前根据策略约束和历史行为审核每项行动提案，验证失败时升级至分析师。这体现了记忆隔离（§ 3.2.2），其中验证层可防止受损上下文导致不可逆的结果。

研究与自适应改进。 报告代理协调自适应侦察与事件后改进。在侦察方面，代理在情景记忆中跟踪假设演化，同时查询威胁情报源和第三方服务。在改进方面，代理提议更新检测规则、剧本和合规映射；改进循环 在提交至组织记忆（§ 3.2.1）之前，根据策略和历史基线审核每项提案。这一双重工作流同时涉及两个表面：工具编排负责外部协调，而内存管理则守护写入持久化知识的路径。

4.2.4 组织记忆管理。运营记忆由威胁知识库、配置管理数据库、安全信息和事件管理数据以及众多数据源构成，只能由独立智能体通过网关访问。这确保了同步和访问控制。对于同步（§3.2.1），更新检测规则、行动后报告或策略库等写操作，均通过报告服务器的改进循环进行，从而提供一条统一的可审计写入路径，防止未经验证的智能体执行写入。版本化存储库和仅追加结构可在整个事件生命周期中保持来源追溯。对于隔离（§3.2.2），访问权限按阶段划分，不存在拥有不受限访问权限的智能体，从而限制影响范围。

4.2.5 人在回路。我们的框架通过手动触发器和直接的端到端监督，确保 SOC 分析师保有决策权。此外，任何阶段中未通过的共识验证都会提交人工审查。这确保框架依赖人工判断，而非在发生故障时自主处置，体现了运营问责要求；智能体增强分析师的专业能力，确保智能体在高风险决策中辅助分析师，而不是取代人工判断。

4.3 AgenticCyOps 框架评估

为评估 AgenticCyOps 的防御覆盖范围，我们提出三项互补分析：用于量化防御原则与攻击向量映射关系的覆盖矩阵 、追踪架构中多步骤攻击链的攻击路径分析 ，以及将可利用攻击面与扁平多智能体系统基线进行比较的信任边界缩减 。

4.3.1 覆盖矩阵。 表3将表1中的每种攻击向量与 AgenticCyOps 的五项防御原则进行交叉对照。当核心机制直接缓解该攻击向量时，单元格标记为（✓）；当其提供次级遏制时，标记为（○）。我们验证了每种攻击向量至少由两项原则加以应对，并且

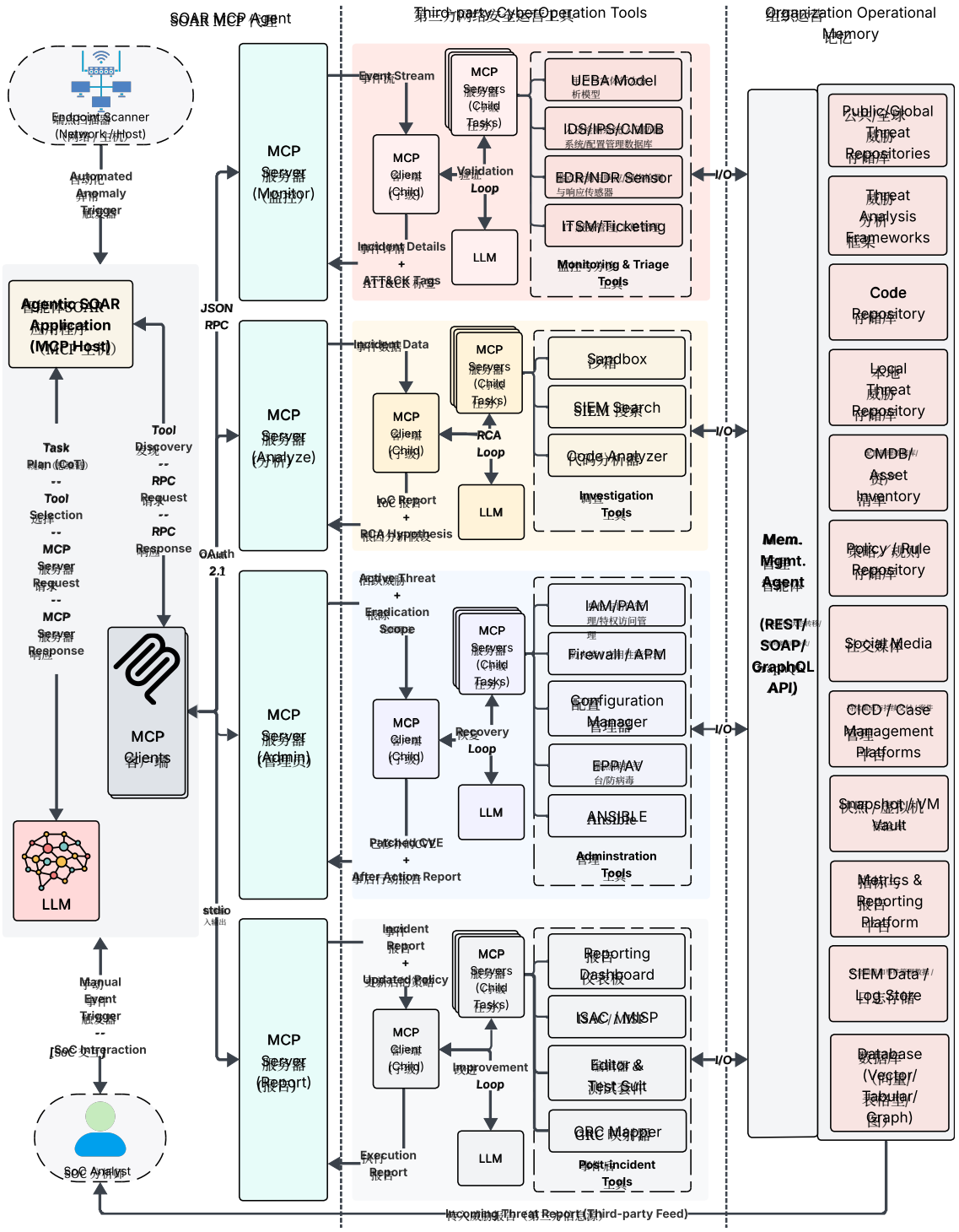

图 4：遵循我们的 AgenticCyOps 框架构建的多智能体 SOAR。该图展示了一种三支柱架构，旨在通过智能体人工智能自动化并编排 SOC 事件响应。MCP 主机充当中央编排器，协调任务规划、工具选择，以及四个按阶段划分的 MCP 服务器（监控、分析、管理员和报告）之间的智能体通信；每个服务器都配备了针对相应生命周期阶段定制的大语言模型驱动的推理循环。持久化的组织记忆层可通过专用的记忆管理智能体访问，为所有智能体提供共享上下文状态，从而在整个事件响应生命周期中确保工作的连续性并支持明智决策。

没有任何单一原则覆盖八种攻击向量中的三种以上。这证实了这五项原则具有互补性：移除其中任何一项，至少会有三种攻击向量仅受到单层防护。

表3：防御原则对 ×攻击向量的覆盖情况

攻击向量	P1	P2	P3	P4	P5
组件级					
未经授权的访问	✓				✓
上下文污染		✓		✓	○
协调层级					
横向入侵		✓	○		✓
共识操纵	✓		✓	✓	
隐蔽协调		✓	✓		○
协议层级					
身份验证绕过	✓				✓
消息操纵			✓	✓	
困惑代理	○	✓	✓		
摘要					
主要（✓）	3	3	3	3	3
次要（○）	1	0	1	0	2

P1=授权接口，P2=能力范围限定，P3=经验证执行，P4=完整性与同步，P5=带数据隔离的访问控制。✓ = 主要缓解措施；○ = 次级遏制。

4.3.2 攻击路径分析。我们通过 CyberOps 架构（图 4）选取覆盖全部三个抽象层并涉及两个集成界面的链路，以确保涵盖框架威胁模型，并确定 AgenticCyOps 的原则在何处对其进行拦截。表 4 总结了每条链路在启用和未启用 AgenticCyOps 强制执行时的情况。在四个场景中的三个（AP-1、AP-2、AP-3）中，AgenticCyOps 在前两步内拦截链路，阻止其升级为利用。每次拦截至少涉及两个作用于不同层的原则，体现了纵深防御。AP-4 揭示了一个已知边界：AgenticCyOps 可限制数据外泄范围和出站模式，但无法强制验证接收组织的摄取管道。

4.3.3 信任边界缩减。 只要一个组件接受来自另一个组件的输入，却未进行独立验证，两者之间就存在信任边界。在扁平多智能体系统部署中，所有智能体都对所有工具和记忆拥有不受限制的访问权限，可利用的边界会呈组合式增长。表 5 对比了有无 AgenticCyOps 时 CyberOps 架构（图 4）的边界数量。在扁平基线中，每个智能体都能访问所有工具（4×16=64）、存储库（4×12=48）和对等节点（4×3=12）。我们的评估（参见附录 C）表明，通过阶段范围限定、主机中介通信和记忆管理智能体仲裁，AgenticCyOps 将可利用的信任边界至少缩减*72%*（200→56）。剩余的 56 个边界并非默认可信：每个边界都至少受到一种主动验证机制（签名清单、共识验证或写入边界过滤）的约束，从而将隐式信任假设转化为明确执行的检查点。

5 讨论（未来展望）

保护工具和记忆表面可应对文献记载的多智能体系统攻击向量，但仍有若干局限值得讨论。垂直架构提供了一个中心信任锚点，却也引入了单点故障；联邦式设计若采用跨主机证明，或可兼顾集中式强制执行与分布式韧性但这些设计仍有待探索 [2]。先验证范式会带来验证延迟，可能影响时间关键型场景，因此需要采用自适应共识阈值，根据行动的可逆性调整严谨程度 [46, 50]。此外，共识循环本身也是攻击面：遭入侵的验证器或被投毒的策略仓库都可能使验证层失效 [55]。跨组织信息源验证不在 AgenticCyOps 的信任边界内（表 4，AP-4）；鉴于 OAuth 2.1 不适用于长期运行的智能体会话，该框架也缺乏用于自主凭证生命周期管理的标准化智能体授权机制 [41]。我们的评估依赖结构分析，而非对抗性模拟，且尚未测量运行时开销；此外，该领域也缺乏用于多智能体安全评估的标准化数据集和基准测试。优先研究方向包括：在拜占庭条件下对共识完整性进行形式化验证、在运营安全运营中心工作负载下开展实证延迟分析、建立多智能体安全基准测试，以及扩展至联邦式拓扑结构。

结论

我们提出了 AgenticCyOps，这是一种面向 MAS 的安全框架，其基础观察是：已有记录的攻击向量集中于两个集成界面：工具编排和记忆管理。五项防御原则与既有合规标准保持一致，为组件层、协调层和协议层提供全面的威胁覆盖，并在 CyberOps 安全编排、自动化与响应架构中得以实现，将安全作为 SOC生命周期中的架构约束。通过覆盖分析、攻击路径追踪和信任边界评估进行的验证表明，可利用边界至少缩减 72%。仍有一些开放性挑战：单点故障韧性、对抗性条件下的共识完整性，以及时间关键型响应中的验证延迟。未来研究的重要方向包括：在生产级企业环境中，使用涵盖多样化用例的标准化数据集开展实证分析，以及进行基准测试开发。

致谢

本研究由阿拉巴马大学计算机科学系的 PATENT实验室开展。本文所表达的观点和结论仅代表作者意见，并不一定代表其所属机构的官方政策或立场。作者还谨此致谢 Flaticon 在图示中的使用。

参考文献

[1] Deepak Bhaskar Acharya、Karthigeyan Kuppan 和 B Divya。2025年。智能体人工智能：面向复杂目标的自主智能——全面综述。*IEEE Access* (2025年)。[2] Apoorva Adimulam、Rajesh Gupta 和 Sumit Kumar。2026。多智能体系统的编排：架构、协议与企业采用。*arXiv*预印本 *arXiv:2601.13671* (2026)。[3] Kushal Agrawal、Verona Teo、Juan J Vazquez、Sudarsh Kunnavakkam、Vishak Srikanth 和 Andy Liu。2025年。评估双重拍卖中的大语言模型智能体串谋。*arXiv*预印本 *arXiv:2507.01413* (2025年)。[4] Zeynab Anbiaee、Mahdi Rabbani、Mansur Mirani、Gunjan Piya、Igor Opushnyev、Ali Ghorbani 和 Sajjad Dadkhah。2026。新兴人工智能智能体协议的安全威胁建模：MCP 、A2A、Agora 和 ANP 的比较分析。*arXiv*预印本 *arXiv:2602.11327* (2026)。[5] Manos Antonakakis、Tim April、Michael Bailey、Matt Bernhard、Elie Bursztein、Jaime Cochran、Zakir Durumeric、J Alex Halderman、Luca Invernizzi、Michalis Kallitsis 等。2017。了解Mirai僵尸网络。载于第*26*届 *USENIX*安全研讨会（*USENIX Security 17*）。1093–1110。

表 4：攻击路径分析：使用和未使用 AgenticCyOps 的代表性攻击链

场景	攻击链	拦截点	AgenticCyOps 响应	残余风险
AP-1： 工具重定向-tion	① 警报源中的对抗性载荷 → ② 监控智能体构造恶意 RPC 请求 → ③ IAM/PAM 工具执行权限提升 → ④ 横向移动至管理员工具	步骤 ②：已阻止	**P2**（能力范围限定）：监控服务器的清单不包含身份与访问管理/特权访问管理工具；主机拒绝了远程过程调用请求。**P3**（经验证执行）：共识循环标记了不同步的工具调用。	已遭入侵 Host
AP-2： 记忆投-毒与证据篡改	①受感染的分析代理将伪造的根因分析写入本地威胁存储库 → ② 管理员智能体读取了被污染的假设-设 → ③ 部署了错误的修复 → ④ 攻击者持久化得以维持	步骤 ①：已阻止	**P4**（内存完整性）：写入边界过滤会验证根因分析输出与事件证据是否一致。**P5**（数据隔离）：跨阶段写入需要记忆管理智能体进行调解。	遭污染的 策略存储库
AP-3： 困惑代理（通过伪造 MCP 响应	① 攻击者拦截 MCP 服务器响应 → ② 伪造响应指示主机删除防火墙规则 → ③ 管理员工具执行破坏性操作 → ④ 边界突破	步骤 ②：已阻止	**P1**（授权接口）：通过 JSON-RPC 消息完整性签名清单检测伪造。**P3**（经验证执行）：重新-covery 循环共识要求多智能体一致后才能执行破坏性操作。	MCP 协议漏洞
AP-4： 跨阶段通过共享上下文	① 报告智能体窃取敏感事件数据 → ② 将武器化指标注入信息共享与分析中心/恶意软件信息共享平台信息流 → ③ 合作组织摄入被污染的入侵指标 → ④ 误报在各安全运营中心之间级联	步骤 ①：部分阻止	**P5**（数据隔离）：报告智能体只能读取事后事件摘要，不能读取原始取证数据。**P2**（能力范围限定）：对 ISAC/MISP 的写入权限仅限于获批指标模式。	跨组织 feed 验证（超出范围限定）

表 5：边界分析：扁平多智能体系统与 AgenticCyOps

边界类型	Flat	AgenticCyOps	Red.
智能体 →工具	64	16	75%
智能体 →记忆	48	16	67%
智能体 ↔智能体	12	4	67%
工具响应 →智能体	64	16	75%
外部信息源 →记忆	12	4	67%
总计	**200**	**56**	**72%**

扁平：每个智能体都可以访问所有工具、存储库和对等节点。AgenticCyOps：阶段范围访问（P2）、主机中介通信、记忆管理智能体中介访问（P5）、共识验证响应（P3）。我们将工具响应计为独立边界，因为返回载荷可能携带注入内容，且不受出站请求授权的约束。

[6] Ido Assaf。2025年。安全运营中心倦怠，以及人工智能如何扭转局面。https://radiantsecurity.ai/blog/soc-burnout-and-how-ai-can-flip-the-script/ 访问日期：2026-02-19。[7] Nicholas Carlini、Florian Tramer、Eric Wallace、Matthew Jagielski、Ariel Herbert-Voss、Katherine Lee、Adam Roberts、Tom Brown、Dawn Song、Ulfar Erlingsson 等。2021年。从大语言模型中提取训练数据。载于第30届*USENIX*安全研讨会（*USENIX Security 21*）。2633–2650。[8] Brian Challita 和 Pierre Parrend。2025年。RedTeamLLM：用于进攻性安全的智能体人工智能框架。arXiv: 2505.06913 [ ] 2505.06913。cs.CR https://arxiv.org/。[9] Gaolei Li、Xi Lin、Zheng Wang 和 Jianhua Li。无日期。BlockAgents：通过区块链实现拜占庭鲁棒的基于大语言模型的多智能体协调。载于*2024*年*ACM*图灵奖庆典会议（中国）论文集。187–192。[10] Zhaorun Chen、Zhen Xiang、Chaowei Xiao、Dawn Song 和 Bo Li。2024年。Agent-poison：通过投毒记忆或知识库对大语言模型智能体进行红队测试。神经信息处理系统进展37 (2024年)，130185–130213。[11] Giulio De Pasquale、Ilya Grishchenko、Riccardo Iesari、Gabriel Pizarro、Lorenzo Cavallaro、Christopher Kruegel 和 Giovanni Vigna。2024年。{ChainReactor}：通过{人工智能}规划自动发现权限提升链。载于第33届*USENIX*安全研讨会（*USENIX Security 24*）。5913–5929。[12] Christian Schroeder de Witt。2025年。多智能体安全面临的开放挑战：迈向由交互式人工智能智能体构成的安全系统。*arXiv*预印本 *arXiv:2505.02077* (2025年)。[13] John R Douceur。2002年。女巫攻击。载于点对点系统国际研讨会。Springer，251–260。[14] Richard Fang、Rohan Bindu、Akul Gupta、Qiusi Zhan 和 Daniel Kang。2024年。大语言模型智能体能够自主入侵网站。*arXiv*预印本 *arXiv:2402.06664* (2024年)。[15] Gartner。2025年。Gartner预测，到2026年，40%的企业应用将采用特定任务人工智能智能体，高于2025年的不到5%。https://www.gartner.com/en/newsroom/press-releases/2025-08-26-gartner-predicts-40-percent-of-enterprise-apps-will-feature-task-specific-ai-agents-by-2026-up-from-less-than-5-percent-in-2025。访问日期：2026-02-20。[16] Google DeepMind。2024年。智能体间（A2A）协议规范：人工智能智能体互操作性的标准化框架。https://github.com/google/a2a-protocol 访问日期：2026-02-16。[17] Kai Greshake、Sahar Abdelnabi、Shailesh Mishra、Christoph Endres、Thorsten Holz 和 Mario Fritz。2023年。并非你所预期：通过间接提示注入攻陷现实世界中的大语言模型集成应用。载于论文集第*16*届美国计算机协会人工智能与安全研讨会论文集. 79–90。[18] Taicheng Guo、Xiuying Chen、Yaqi Wang、Ruidi Chang、Shichao Pei、Nitesh V Chawla、Olaf Wiest 和 Xiangliang Zhang。2024。基于大语言模型的多智能体：进展与挑战综述。*arXiv* 预印本 *arXiv:2402.01680*（2024）。[19] William GJ Halfond、Jeremy Viegas、Alessandro Orso 等。2006。SQL 注入攻击及其对策分类。（2006）。[20] Pengfei He、Yuping Lin、Shen Dong、Han Xu、Yue Xing 和 Hui Liu。2025。通过通信攻击对大语言模型多智能体系统进行红队测试。载于计算语言学协会年会论文集：*ACL 2025*。6726–6747。[21] IBM。2024。什么是智能体架构？IBM 公司。ibm.com/think/topics/agentic-architecture IBM Think 主题。[22] IBM 公司。2024。人工智能智能体的组件。网络文章。ibm.com/think/topics/components-of-ai-agents IBM Think。[23] IBM 公司。2024。IBM SOAR。ibm.com/products/soar-platform 安全编排、自动化与响应平台。[24] Ismail、Rahmat Kurnia、Zilmas Arjuna Brata、Ghitha Afina Nelistiani、Shinwook Heo、Hyeongon Kim 和 Howon Kim。2025。采用智能体人工智能超自动化方法，推动安全运营中心实现稳健的安全编排与自动化响应。《Information》16，5（2025），365。[25] Yujie Ji、Xinyang Zhang 和 Ting Wang。2017。针对学习系统的后门攻击。载于*2017*年电气电子工程师学会通信与网络安全会议（*CNS*）。电气电子工程师学会，1–9。[26] Mohammad Sami A. Kabir。2023。从孟加拉国银行盗窃案中汲取的教训。国际信息系统审计协会期刊6（2023）。https://www.isaca.org/resources/isaca-journal/issues/2023/volume-6/lessons-learned-from-the-bangladesh-bank-heist [27] Shaharyar Khan、Ilya Kabanov、Yunke Hua 和 Stuart Madnick。2022。对第一资本数据泄露事件的系统分析：重要经验教训。《ACM 隐私与安全汇刊》26，1（2022），1–29。[28] Juhee Kim、Woohyuk Choi 和 Byoungyoung Lee。 [n. d.]。大语言模型智能体中的提示流完整性：防止权限提升。*arXiv* 预印本 *arXiv:2503.15547*（[n.d.]）。[29] Klaudia Krawiecka 和 Christian Schroeder de Witt。2025。扩展开放式全球应用程序安全项目多智能体系统威胁建模指南：多智能体安全研究的见解。*arXiv* 预印本 *arXiv:2508.09815*（2025）。[30] Matteo Lupinacci、Francesco Aurelio Pironti、Francesco Blefari、Francesco Romeo、Luigi Arena 和 Angelo Furfaro。2025。大语言模型的阴暗面：可完全接管计算机的基于智能体的攻击。*arXiv* 预印本 *arXiv:2507.06850*（2025）。[31] Sean Lyngaas。2024。Crowdstrike 报告：网络攻击的平均“突破时间”缩短至62分钟。CyberScoop。https://cyberscoop.com/crowdstrike-annual-global-threat-report-attack-breakout-time/ 访问日期：2026年2月27日。[32] Junyuan Mao、Fanci Meng、Yifan Duan、Miao Yu、Xiaojun Jia、Junfeng Fang、Yuxuan Liang、Kun Wang 和 Qingsong Wen。2025。AgentSafe：通过分层数据管理保护基于大语言模型的多智能体系统。*arXiv* 预印本 *arXiv:2503.04392*（2025）。[33] Yohan Mathew、Ollie Matthews、Robert McCarthy、Joan Velja、Christian Schroeder de Witt、Dylan Cope 和 Nandi Schoots。2025。隐藏在明文之中：大语言模型中隐写串通的出现与缓解。载于第*14*届自然语言处理国际联合会议暨计算语言学协会亚太分会第*4*届会议论文集。585–624。[34] Shaswata Mitra、Azim Bazarov、Martin Duclos、Sudip Mittal、Aritran Piplai、Md Rayhanur Rahman、Edward Zieglar 和 Shahram Rahimi。2025。FALCON：使用大语言模型自主挖掘网络威胁情报以生成入侵检测系统规则。arXiv:2508.18684 [cs.CR] https://arxiv.org/abs/2508.18684

[35] Shaswata Mitra、Subash Neupane、Trisha Chakraborty、Sudip Mittal、Aritran Piplai、Manas Gaur 和 Shahram Rahimi。2024。LocalIntel：从全球和本地网络安全知识中生成组织威胁情报。载于国际安全基础与实践研讨会。Springer，63–78。[36] [在线] MITRE。2011。MITRE ATT&CK。https://attack.mitre.org/。[37] 模型上下文协议。2025。架构——模型上下文协议。https: //modelcontextprotocol.io/specification/2025-06-18/architecture 规范版本 2025-06-18。[38] Vineeth Sai Narajala 和 Idan Habler。2025。模型上下文协议（MCP）的企业级安全：框架与缓解策略。arXiv 预印本 arXiv:2504.08623（2025）。[39] Vineeth Sai Narajala、Ken Huang 和 Idan Habler。2025。通过基于零信任注册表的方法，防范针对生成式人工智能多智能体系统的工具蹲占。arXiv 预印本 arXiv:2504.19951（2025）。[40] [在线] NIST。2014。NIST。nist.gov/cyberframework/online-learning/five-functions。[41] OWASP 生成式人工智能安全项目。2025。OWASP 智能体人工智能应用十大风险：ASI03——身份与权限滥用。https://genai.owasp.org/ 版本 1.0。[42] Carlota Perez。2002。技术革命与金融资本：泡沫与黄金时代的动态。载于《技术革命与金融资本》。Edward Elgar Publishing。[43] Ethan Perez、Saffron Huang、Francis Song、Trevor Cai、Roman Ring、John Aslanides、Amelia Glaese、Nat McAleese 和 Geoffrey Irving。2022。使用语言模型对语言模型进行红队测试。载于《2022年自然语言处理实证方法会议论文集》。3419–3448。[44] Alireza Rezazadeh、Zichao Li、Ange Lou、Yuying Zhao、Wei Wei 和 Yujia Bao。2025。协作记忆：通过动态访问控制实现大语言模型智能体的多用户记忆共享。arXiv 预印本 arXiv:2505.18279（2025）。[45] Yucheng Shi、Wenhao Yu、Wenlin Yao、Wenhu Chen 和 Ninghao Liu。2025。迈向可信的图形用户界面智能体：综述。arXiv 预印本 arXiv:2503.23434（2025）。[46] Madhulika Srikumar、Jacob Pratt、Kasia Chmielinski、Carolyn Ashurst、Chloé Bakalar、William Bartholomew、Rishi Bommasani、Peter Cihon、Rebecca Crootof、Mia Hoffmann 等。2025。优先实现人工智能智能体的实时故障检测。人工智能合作组织（2025）。[47] Saksham Sahai Srivastava 和 Haoyu He。2025。MemoryGraft：通过投毒经验检索对大语言模型智能体进行持续性入侵。arXiv 预印本 arXiv:2512.16962（2025）。[48] Georgios Syros、Anshuman Suri、Jacob Ginesin、Cristina Nita-Rotaru 和 Alina Oprea。2025。SAGA：用于治理人工智能智能体系统的安全架构。arXiv 预印本 arXiv:2504.21034（2025）。[49] 人工智能法案。2025。《欧盟人工智能法案》：最新进展与分析。https://artificialintelligenceact.eu/ 访问日期：2026-02-17。[50] Nenad Tomašev、Matija Franklin 和 Simon Osindero。2026。智能人工智能委派。arXiv 预印本 arXiv:2602.11865（2026）。[51] Total Assure。2025。2025年网络攻击平均检测时间：每家企业都必须了解的关键检测统计数据。Total Assure 博客。totalassure.com/blog/average-time-to-detect-cyber-attack-2025 访问日期：2025-02-27。[52] Harold Triedman、Rishi Jha 和 Vitaly Shmatikov。2025。多智能体系统会执行任意恶意代码。arXiv 预印本 arXiv:2503.12188（2025）。[53] James Tu、Tsunhsuan Wang、Jingkang Wang、Sivabalan Manivasagam、Mengye Ren 和 Raquel Urtasun。2021。针对多智能体通信的对抗攻击。载于电气电子工程师学会/计算机视觉基金会国际计算机视觉会议论文集。7768–7777。[54] Bo Wang、Weiyi He、Shenglai Zeng、Zhen Xiang、Yue Xing、Jiliang Tang 和 Pengfei He。2025。揭示大语言模型智能体记忆中的隐私风险。载于计算语言学协会第63届年会论文集（第1卷：长篇论文）。25241–25260。[55] Qianshan Wei、Tengchao Yang、Yaochen Wang、Xinfeng Li、Lijun Li、Zhenfei Yin、Yi Zhan、Thorsten Holz、Zhiqiang Lin 和 XiaoFeng Wang。2025。A-MemGuard：面向基于大语言模型的智能体记忆的主动防御框架。arXiv 预印本 arXiv:2510.02373（2025）。[56] Shanglin Wu 和 Kai Shu。2025。大语言模型多智能体系统中的记忆：机制、挑战与集体智能。Authorea 预印本（2025）。[57] Chong Xiang、Tong Wu、Zexuan Zhong、David Wagner、Danqi Chen 和 Prateek Mittal。2024。针对检索污染的可认证鲁棒检索增强生成。arXiv 预印本 arXiv:2405.15556（2024）。[58] Faouzi El Yagoubi、Ranwa Al Mallah 和 Godwin Badu-Marfo。2026。AgentLeak：用于评估多智能体大语言模型系统隐私泄露的全栈基准测试。arXiv 预印本 arXiv:2602.11510（2026）。[59] Sizhe Yuen、Francisco Gomez Medina、Ting Su、Yali Du 和 Adam J Sobey。2025。内在记忆智能体：通过结构化上下文记忆构建异构多智能体大语言模型系统。arXiv 预印本 arXiv:2508.08997（2025）。[60] Zhenhua Zou、Zhuotao Liu、Lepeng Zhao 和 Qiuyang Zhan。2025。BlockA2A：迈向安全、可验证的智能体间互操作性。arXiv 预印本 arXiv:2508.01332（2025）。

A 附录 - 首字母缩略词列表

A2A 智能体间协议 AAR 事后行动报告 ABAC 基于属性的访问控制 AI 人工智能 API 应用程序编程接口 APM 访问策略管理器 AV 防病毒 BCP 业务连续性计划 CMDB 配置管理数据库 CMS 配置管理系统 CoT 思维链 CTI 网络威胁情报 CVE 通用漏洞与暴露 CyberOps 网络安全运营 EDR 端点检测与响应 EPP 端点保护平台 EU 欧盟 GDPR 通用数据保护条例 GraphQL 图查询语言 GRC 治理、风险与合规 I/O 输入/输出 IAM 身份与访问管理 IDS 入侵检测系统 IoC 失陷指标 IoT 物联网 IPS 入侵防御系统 ISAC 信息共享与分析中心 ISO 国际标准化组织 ITSM 信息技术服务管理 JSON JavaScript 对象表示法 LLM 大语言模型 MAS 多智能体系统 MCP 模型上下文协议 MISP 恶意软件信息共享平台 ML 机器学习 MTTD 平均检测时间 MTTR 平均响应时间 NDR 网络检测与响应 NIST 国家标准与技术研究院 NVD 国家漏洞数据库 OAuth 开放授权 OSINT 开源情报 OWASP 开放式全球应用程序安全项目 PAM 特权访问管理 PFI 提示流完整性 RAG 检索增强生成 RBAC 基于角色的访问控制 RCA 根因分析 REST 表述性状态转移 RPC 远程过程调用 SIEM 安全信息与事件管理 SOAP 简单对象访问协议 SOAR 安全编排、自动化与响应

e

SOC 安全运营中心 SP 特别出版物（NIST） SQL 结构化查询语言 TTP 战术、技术与程序 UEBA 用户与实体行为分析 XSS 跨站脚本攻击

B 附录 – 文献基础

下表总结了构建 AgenticCyOps 攻击面分解和防御框架所依据的主要文献。对于每项研究，我们说明其对分析的贡献、范围限定，以及对框架设计具有指导意义的具体要点。

表 6：文献基础：攻击面识别、防御机制与网络运营领域

参考文献	贡献	范围／局限性	AgenticCyOps 启示	对应
组件级攻击面（*§2.1*）				
Greshake 等人 [17]	通过检索到的外部数据实施间接提示注入；在 Bing Chat 和 GitHub 上验证 Copilot。	单智能体流水线；无多智能体传播。	模糊 数据/指令 边界，启用工具重定向和凭证数据外泄；促使编写边界过滤。	表1：上下文污染
Perez 等人 [43]	通过自动化红队测试诱导有害的大语言模型通过对抗性提示产生的输出。	仅涉及模型级鲁棒性；不涉及集成层架构。	认知操纵不在范围限定之内；集成层防御应对下游影响。	模型级范围排除（§2.1）
Carlini等人 [7]	通过定向查询从大语言模型中提取训练数据。	单模型隐私风险；代理之间没有共享内存。	通过记忆隔离来防止跨智能体数据泄露。	P5：访问控制隔离
Fang 等人 [14]	大语言模型智能体自主攻击网站 利用 SQL 注入和跨站脚本攻击；GPT-4 的成功率达到 73.3%。功率。	攻击性研究；没有防御措施，也未涉及多智能体工具滥用。	自主攻击性工具调用促使人们限定能力范围并落实最小权限原则。	P2：能力范围限定
协调层攻击面（*§2.2*）				
Chen 等人 [9]	BlockAgents：采用拜占庭鲁棒多智能体系统的区块链思维证明共识以抵御串谋和投毒。	仅限共识层；无工具编排或记忆隔离。	多智能体系统中的拜占庭式串谋促使我们采用共识验证和经过验证的执行。	表1：横向入侵
Douceur [13]	通过伪造身份形式化描述去中心化系-统中的女巫攻击。统中的女巫攻击。	经典分布式系统；并非针对智能体-特定场景。	通过伪造身份实施女巫攻击-形成智能体接口处的身份绑定和身份验证智能体接口处的..... 二四六	P1：授权接口
Agrawal 等人 [3]	前沿大语言模型中的涌现式串通，无需显式提示。	评估性；未提出防御机制提出。	串谋作为协调层威胁促使开展能力审计和共-识验证。	表1：隐蔽协调
Mathew 等人 [33]	隐写串通会无意间出现，源于大语言模型的奖励激励设定不当，从而绕过监督。	侧重检测；编码方案有限。	涌现的隐蔽信道绕过传统监控；这促使人们采用内存隔离和输出过滤。	表1：隐蔽协调
协议级攻击面（*§2.3*）				
Triedman 等人 [52]	通过智能体间 信任利用劫持多智能体系统控制流；已在 AutoGen、CrewAI, MetaGPT 上验证。	多智能体系统专属；没有记忆向量或防御机制。	通过混淆 代理利用促使我们采用经授权接口和经过验证的执行过程。	P1、P3；表1：困惑代理
De Pasquale 等人 [12]	多智能体威胁分类法：秘密协合谋、级联攻击和利用。	广泛的分类；没有统一的防御模型或验证。	多智能体威胁分类推动了——跨协调与协议层面的纵深防御——国家和协议层面。	表1：协议级攻击向量
Shi 等 [45]	GUI 智能体安全综述；指出对抗性输入漏洞，并提出输入验证和沙箱隔离。输入验证和沙箱隔离。	GUI 智能体特有；不涉及多智能体系统协调..... tion 或记忆向量。	感知边界处的对抗性输入-ary 促使写入边界过滤-ing 以及授权接口强制执行-ment。	P1：授权接口
He 等人 [20]	中间代理攻击（AiTM）攻击截获并操纵智能体间通信；在 MetaGPT 和 ChatDev 上进行了验证。	侧重通信；防御有限，未涉及记忆或工具向量。	智能体间消息操纵促使经验证执行和授权的接口强制执行。	表1：消息操纵
开放式全球应用程序安全项目 [41]	文档列出了主要的智能体人工智能风险，包身份/特权滥用和过度代理；指出 OAuth 2.1 在长时间运行的会话方面存在不足。	标准差距分析；没有其他授权框架。	身份与权限滥用促使智能体专用接口授权——与能力范围限定。	P1：授权接口
Anbiaee 等人 [4]	对 MCP、A2A、Agora 和 ANP 进行比较威胁建模；识别出十二种协议——整个生命周期阶段的风险。	仅有风险分类体系；没有防御架构或实证验证。	跨协议风险碎片化验证了协议无关的防御设计适用于 AgenticCyOps。	表1：协议级攻击向量
工具集成防御措施（*§3.1.1–§3.1.3*）				
Narajala & Sridhar [39]	工具蹲占威胁与零信任工具注册表，包含管理员验证的目录和即时凭证配置。	架构提案；未经实证验证。	工具蹲占促使采用签名清单，以及管理员批准的目录 P1.	P1：授权接口
Narajala & Sridhar [38]	采用零信任的 MCP 安全框架，应对工具投毒、数据外泄以及通过密码学签名和严格输入输出实现命令与控制验证。	专用于 MCP；实现复杂度限制了采用。	MCP 威胁态势与零信任缓解措施促使采用密码学 P1 中的来源追溯。	P1：授权接口
Syros 等人 [48]	SAGA：密码学身份绑定，用户-受控的智能体生命周期，以及基于令牌的智能体间通信；在八种攻击者模型下进行评估。	集中式架构；无记忆隔离。	密码学身份绑定和基于令牌的访问控制补充 P1 中基于注册表的发现。	P1：授权接口
Kim 等人 [28]	PFI：通过智能体隔离、不可信数据防护机制和最低限度的权限执行；在 Agent-Dojo 和 AgentBench 上进行了评估。	单智能体；仅限于提示和数据注入向量。	PFI 的最小权限执行和隔离防护措施直接为 P2 提供依据能力范围限定。	P2：能力范围限定
Krawiecka 等 [29]	扩展了 OWASP MAS 威胁建模指南；指出不安全的委派升级和涌现式隐蔽协同。	分类体系扩展；没有实证验证或强制执行机制。	不安全的委派升级促使能力范围限定和持续特权审计。	P2：能力范围限定
De Pasquale 等人 [11]	ChainReactor：自动化权限提升链发现；在真实环境中发现了 20 条新链——AWS EC2 和 Digital Ocean 实例。	攻击性工具；不涉及初始渗透或漏洞利用代码生成。	自动化权限提升发现促使人们进行能力范围限定并执行最小权限。	P2：能力范围限定

续见下页

参考文献	贡献	范围／局限性	AgenticCyOps 启示	对应
Zou 等人 [60]	BlockA2A：三层信任框架（身- 份、账本、智能合约），采用去中心- 化身份、不可篡改的溯源以及可编程- 治理。	区块链开销；链上操- 操作需要进行批处理。	去中心化身份绑定和不可变审计轨迹为经验证执- 行与不可否认性提供依据，适用于 P3。 行与不可否认性。	P3：经验证执行
内存管理防御措施（*§3.2.1–§3.2.2*）				
Wei 等人 [55]	A-MemGuard：基于共识的验证 以及用于检测内存的双内存结构 大语言模型智能体中的注入攻击。	主要为单智能体；多智能体 可扩展性尚未充分探索。	基于共识的内存验证 和双内存结构为……提供依据 写入边界过滤和投毒 检测，为 P4 提供依据。	P4：内存完整性
Mao 等人 [32]	AgentSafe：ThreatSieve for 权限- 受控通信与 HierarCache 用于分层内存隔离；在 多种拓扑结构中进行了评估。	模拟环境；无真实 部署验证。	分层内存隔离以及 权限控制的 通信 通知 P4 完整性和 P5 隔离。	P4、 P5
Xiang 等人 [57]	RobustRAG：可认证鲁棒检索增强生成 via 结合关键词的先隔离后聚合，以及 针对检索的基于解码的聚合 破坏。	有界 对手； 有限 to 段落级破坏和单- 跳检索。	采用先隔离后聚合和可认证鲁棒性 进行共识验证的 P4 中的检索。	P4：内存完整性
Chen 等人 [10]	AGENTPOISON：后门攻击优化- 利用隐蔽触发器对基于RAG的 智能体记忆进行投毒，同时保持良性性能 表现。	侧重攻击；防御方案有限 。	对RAG进行后门触发器投毒 记忆促使在写入边界进行过 滤，并在 P4 中进行来源追踪。	P4：内存完整性
Wu等人 [56]	记忆机制、架构以及 大语言模型中的一致性挑战 多智能体系统涵盖拓扑模式及同步 协议。	仅为综述；不提供持久记忆 框架建议。	跨拓扑结构的记忆一致性协议 共享/本地拓扑促使人们认真 P4 中的标准化和版本控制。	P4：内存完整性
Wang等人 [54]	MEXTRA：黑盒记忆提取，使用 定位器-对齐器提示来提取私有 大语言模型智能体记忆中的查询。	单智能体；无多智能体泄露 或会话控制。	黑盒记忆提取促- 使严格的访问区隔- 化及 P5 中的记忆净化。	P5：访问控制隔离
Yagoubi 等人 [58]	AgentLeak：全栈基准测试，用于评估 多智能体系统通道中的隐私泄露，涵盖 1,000 个场景和 32 类分类体系。	场景覆盖范围、语言和 拓扑范围有限。	内部通道泄露和默认- 开放共享促使记忆访问 控制与默认拒绝隔离在 P5.	P5：访问控制隔离
Rezazadeh等人 [44]	协作记忆：多用户多智能体系统与 动态二分访问图、双层私有/ 共享内存，以及细粒度读写 策略。	受控环境；概率性 大语言模型行为限制了确定性 保证。	动态二分访问图和 分层记忆隔离为访问 控制提供依据，适 用于 P5。	P5：访问控制隔离
Yuen等人 [59]	内在记忆智能体：结构化智能体- 特定的记忆模板，以维护角色 在异构多智能体系统中的相关上下文。	手动创建模板；任务 多样性验证有限。	智能体专属的结构化记忆隔 离可确定角色范围边界 以及 P5 中的访问约束。	P5：访问控制隔离
Srivastava 等人 [47]	MemoryGraft：两阶段攻击，注入有毒的 经验记录到智能体记忆中， 通过语义手段造成持续的行为漂移， 性检索。	单智能体；仅限于汇总检 索统计数据。	无触发器的持久记忆污 染促使人们采用密码学证明 和写入边界共识 在 P4 中。	P4：记忆完整性
网络安全运营领域与架构（*§4*）				
Ismail等人 [24]	使用智能体大语言模型的超自动化SOAR 采用 IVAM生命周期和动态行动手册 生成；将38步剧本缩减为10个 个在 Wazuh 安全信息与事件管理系统上执行的行动。	试点部署；场景有限 且未涉及人工智能可解释性。	智能体SOAR，具备动态剧本调 整及人在回路中的 循环促使在 安全运营中心工作流。	§4.1：运营差距
Mitra 等人 [35]	LocalIntel：结合上下文的网络威胁情报推理，具备 智能体自主性。	单智能体；无多智能体协 调或记忆隔离。	验证了对上下文感知 推理的需求；这些缺口促使构建完整的多智能体系统。	§4.1.1：分析推理，§4.1.3 研究
Mitra 等人 [34]	FALCON：自主策略生成与 迭代式假设完善。	单智能体威胁 狩猎； no 工具/记忆安全模型。	采用自适应侦察；安全 漏洞促使我们构建防御层。	§4.1.2：决策 制定， §4.1.3 自适应改进
Challita 等人 [8]	RedTeamLLM：借助 动态计划修正、记忆管理 和 ReAct 推理实现智能体式进攻安全；表现优于 Pen- testGPT 在 VULNHUB 上的表现。	概念验证；记忆管理 实施和规划纠正未完全实- 施。	具备智能体式进攻能力并采用 自适应规划，验证了对动态操作手 册在网络安全运营中的需求。	§4.1.3：自适应改进
Tomašev 等 [50]	具备要求的智能委托框架， 包括动态评估、适应性执 行、结构透明性、可扩展的 协调和系统韧性； 映射到 MCP、A2A、AP2 和 UCP 协议 协议。	仅限理论层面；没有实证实施或验- 证。	用于信任校- 准和系统韧性的委托要求与 AgenticCyOps 已验证的执行- tion 设计。	第5节：架构对齐
Lupinacci [30]	三种攻击技术；智能体间信任利 用在所有受测模型中均达到 100% 的成功率 ，从而实现计算机接管 通过提示注入和 RAG后门。	合成环境；未提出防御 框架。	通用的智能体间信任利用 以及 RAG后门验证了工具 和内存表面漏洞 跨组件和协调 层。	表1：覆盖率验证

C 附录：TrustBoundaryEnumeration

本附录枚举了扁平（不受限）多智能体系统部署中的全部 200 个信任边界，并确定其中哪些信任边界已被 AgenticCyOps 的阶段范围限定和中介架构消除，以及哪些信任边界在主动验证下保留。该枚举支持表 5 中 72% 的缩减声明（200→56）。

架构参数。

- **代理（4 个）：** A1=监控、A2=分析、A3=管理员、A4=报告
- **工具（16 个）：** T1=用户与实体行为分析模型、T2=入侵检测系统/入侵防御系统/配置管理数据库、T3=端点检测与响应/网络检测与响应传感器、T4=IT 服务管理/工单管理（监控）；T5=沙箱、T6=SIEM 搜索、T7=代码分析器（分析）；T8=身份与访问管理/特权访问管理、T9=防火墙/应用性能管理、T10=配置管理器、T11=端点保护平台/防病毒、T12=Ansible（管理员）；T13=报告仪表板、T14=信息共享与分析中心/恶意软件信息共享平台、T15=编辑器与测试套件、T16=GRC 映射器（报告）
- **内存存储（12）：** M1=威胁存储库，M2=资产清单，M3=策略存储库，M4=安全信息与事件管理数据湖，M5=代码仓库，M6=案件管理，M7=CTI 知识库，M8=剧本仓库，M9=合规映射，M10=检测规则，M11=AAR存档，M12=业务连续性计划/风险登记册
- **外部信息源（12）：** E1=供应商公告，E2=信息共享与分析中心信息源，E3=MISP社区，E4=通用漏洞披露/国家漏洞数据库，E5=开源情报来源，E6=监管更新，E7=威胁情报订阅，E8=漏洞扫描器，E9=合作伙伴安全运营中心信息源，E10=行业基准，E11=补丁公告，E12=合规更新

阶段到工具分配（P2：能力范围界定）。A1→{T1–T4}，A2→{T5–T7}，A3→{T8–T12}，A4→{T13–T16}**阶段到内存分配（P5：数据隔离访问控制）**。A1→{M1,M4,M7,M10}，A2→{M1,M2,M6,M7}，A3→{M2,M3,M8,M12}，A4→{M8,M9,M10,M11}**图例**。×=保留（采用主动验证机制）；　✓=由 AgenticCyOps 消除。

类别 1：智能体 → 工具边界 共 *64* 个，保留 *16* 个，消除 *48* 个（*75%*）阶段范围限定（P2）将每个智能体限制为其分配的工具（监控：4 个，分析：3 个，管理员：5 个，报告：4 个）。保留边界须经过签名清单验证（P1）和共识验证执行（P3）。

#	智能体	Tool	状态	机制
A1（监控器） → 工具 *[4 已分配：T1–T4]*				
1	A1-监控	T1-UEBA 模型	×	P1+P3：已签名清单、验证循环
2	A1-监控	T2-入侵检测系统/入侵防御系统/配置管理数据库	×	P1+P3：已签名清单、验证循环
3	A1-监控	T3-EDR/NDR 传感器	×	P1+P3：已签名清单，验证循环
4	A1-监控	T4-ITSM/工单管理	×	P1+P3：已签名清单，验证循环
5	A1-监控	T5-沙箱	✓	P2：不在监控清单中
6	A1-监控	T6-SIEM 搜索	✓	P2：不在监控清单中
7	A1-监控	T7-代码分析器	✓	P2：不在监控清单中
8	A1-监控	T8-IAM/PAM	✓	P2：不在监控清单中
9	A1-监控	T9-防火墙/APM	✓	P2：不在监控清单中
10	A1-监控	T10-配置管理器	✓	P2：不在监控清单中
11	A1-监控	T11-EPP/AV	✓	P2：不在监控清单中
12	A1-监控	T12-ANSIBLE	✓	P2：不在监控清单中
13	A1-监控	T13-报告仪表板	✓	P2：不在监控清单中
14	A1-监控	T14-ISAC/MISP	✓	P2：不在监控清单中
15	A1-监控	T15-编辑器与测试	✓	P2：不在监控清单中
16	A1-监控	T16-GRC 映射器	✓	P2：不在监控清单中
A2（分析） → 工具 *[3已分配：T5–T7]*				
17	A2-分析	T1-UEBA 模型	✓	P2：不在分析清单中
18	A2-分析	T2-入侵检测系统/入侵防御系统/配置管理数据库	✓	P2：不在分析清单中
19	A2-分析	T3-EDR/NDR 传感器	✓	P2：不在分析清单中
20	A2-分析	T4-ITSM/工单管理	✓	P2：不在分析清单中
21	A2-分析	T5-沙箱	×	P1+P3：已签名清单，RCA 循环
22	A2-分析	T6-SIEM 搜索	×	P1+P3：已签名清单，RCA 循环
23	A2-分析	T7-代码分析器	×	P1+P3：已签名清单、RCA 循环
24	A2-分析	T8-IAM/PAM	✓	P2：不在分析清单中
25	A2-分析	T9-防火墙/APM	✓	P2：不在分析清单中
26	A2-分析	T10-配置管理器	✓	P2：不在分析清单中
27	A2-分析	T11-EPP/AV	✓	P2：不在分析清单中
28	A2-分析	T12-ANSIBLE	✓	P2：不在分析清单中
29	A2-分析	T13-报告仪表板	✓	P2：不在分析清单中
30	A2-分析	T14-ISAC/MISP	✓	P2：不在分析清单中
31	A2-分析	T15-编辑器与测试	✓	P2：不在分析清单中
32	A2-分析	T16-GRC 映射器	✓	P2：不在分析清单中
A3（管理员） →工具 *[5已分配：T8–T12]*				
33	A3-管理	T1-UEBA 模型	✓	P2：不在管理员清单中
34	A3-管理	T2-入侵检测系统/入侵防御系统/配置管理数据库	✓	P2：不在管理员清单中
35	A3-管理	T3-EDR/NDR 传感器	✓	P2：不在管理员清单中
36	A3-管理	T4-ITSM/工单管理	✓	P2：不在管理员清单中
37	A3-管理	T5-沙箱	✓	P2：不在管理员清单中
38	A3-管理	T6-SIEM 搜索	✓	P2：不在管理员清单中
39	A3-管理	T7-代码分析器	✓	P2：不在管理员清单中
40	A3-管理	T8-IAM/PAM	×	P1+P3：已签名清单，恢复循环
41	A3-管理	T9-防火墙/APM	×	P1+P3：已签名清单，恢复循环
42	A3-管理	T10-配置管理器	×	P1+P3：已签名清单，恢复循环
43	A3-管理	T11-EPP/AV	×	P1+P3：已签名清单，恢复循环
44	A3-管理	T12-ANSIBLE	×	P1+P3：已签名清单，恢复循环
45	A3-管理	T13-报告仪表板	✓	P2：不在管理员清单中
46	A3-管理	T14-ISAC/MISP	✓	P2：不在管理员清单中
47	A3-管理	T15-编辑器与测试	✓	P2：不在管理员清单中
48	A3-管理	T16-GRC 映射器	✓	P2：不在管理员清单中
A4（报告） →工具 *[4 已分配：T13–T16]*				
49	A4-报告	T1-UEBA 模型	✓	P2：不在报告清单中
50	A4-报告	T2-入侵检测系统/入侵防御系统/配置管理数据库	✓	P2：不在报告清单中

续见下一页

#	智能体	Tool	状态	机制
51	A4-报告	T3-EDR/NDR 传感器	✓	P2：不在报告清单中
52	A4-报告	T4-ITSM/工单管理	✓	P2：不在报告清单中
53	A4-报告	T5-沙箱	✓	P2：不在报告清单中
54	A4-报告	T6-SIEM 搜索	✓	P2：不在报告清单中
55	A4-报告	T7-代码分析器	✓	P2：不在报告清单中
56	A4-报告	T8-IAM/PAM	✓	P2：不在报告清单中
57	A4-报告	T9-防火墙/APM	✓	P2：不在报告清单中
58	A4-报告	T10-配置管理器	✓	P2：不在报告清单中
59	A4-报告	T11-EPP/AV	✓	P2：不在报告清单中
60	A4-报告	T12-ANSIBLE	✓	P2：不在报告清单中
61	A4-报告	T13-报告仪表板	×	P1+P3：已签名清单，改进循环
62	A4-报告	T14-信息共享与分析中心/恶意软件信息共享平台	×	P1+P3：已签名清单，改进循环
63	A4-报告	T15-编辑器与测试	×	P1+P3：已签名清单，改进循环
64	A4-报告	T16-GRC 映射器	×	P1+P3：已签名清单，改进循环

类别 1 摘要： 16 项保留（×），48 项消除（✓）。缩减：75%。

类别 2：智能体 → 内存边界 共计 *48* 项，保留 *16* 项，消除 *32* 项（*67%*）分阶段访问（P5）将每个智能体限制为四个指定的内存存储。所有保留边界均由记忆管理智能体通过写入边界过滤（P4）进行中介。

#	智能体	记忆存储	状态	机制
A1（监控器） → 记忆				
65	A1-监控	M1-威胁存储库	×	P4+P5：MMA介导、写入过滤
66	A1-监控	M2-资产清单	✓	P5：监控范围之外
67	A1-监控	M3-策略存储库	✓	P5：监控范围之外
68	A1-监控	M4-SIEM数据湖	×	P4+P5：MMA介导，写入过滤
69	A1-监控	M5-代码仓库	✓	P5：监控范围之外
70	A1-监控	M6-案例管理	✓	P5：监控范围之外
71	A1-监控	M7-CTI知识库	×	P4+P5：MMA介导，只读
72	A1-监控	M8-剧本仓库	✓	P5：监控范围之外
73	A1-监控	M9-合规映射	✓	P5：监控范围之外
74	A1-监控	M10-检测规则	×	P4+P5：MMA介导，只读
75	A1-监控	M11-AAR档案	✓	P5：监控范围之外
76	A1-监控	M12-BCP/风险登记册	✓	P5：监控范围之外
A2（分析） → 记忆				
77	A2-分析	M1-威胁存储库	×	P4+P5：MMA介导，写入过滤
78	A2-分析	M2-资产清单	×	P4+P5：MMA介导，只读
79	A2-分析	M3-策略存储库	✓	P5：超出分析范围限定
80	A2-分析	M4-SIEM数据湖	✓	P5：超出分析范围限定
81	A2-分析	M5-代码仓库	✓	P5：超出分析范围限定
82	A2-分析	M6-案例管理	×	P4+P5：MMA介导、写入过滤
83	A2-分析	M7-CTI知识库	×	P4+P5：MMA介导，只读
84	A2-分析	M8-剧本仓库	✓	P5：超出分析范围
85	A2-分析	M9-合规映射	✓	P5：超出分析范围
86	A2-分析	M10-检测规则	✓	P5：超出分析范围
87	A2-分析	M11-AAR档案	✓	P5：超出分析范围限定
88	A2-分析	M12-BCP/风险登记册	✓	P5：超出分析范围限定
A3（管理员） →记忆				
89	A3-管理	M1-威胁存储库	✓	P5：不属于管理员范围
90	A3-管理	M2-资产清单	×	P4+P5：MMA介导、写入过滤
91	A3-管理	M3-策略存储库	×	P4+P5：MMA介导、只读
92	A3-管理	M4-SIEM数据湖	✓	P5：不属于管理员范围
93	A3-管理	M5-代码仓库	✓	P5：不属于管理员范围
94	A3-管理	M6-案例管理	✓	P5：不属于管理员范围
95	A3-管理	M7-CTI知识库	✓	P5：不属于管理员范围
96	A3-管理	M8-剧本仓库	×	P4+P5：MMA介导，只读
97	A3-管理	M9-合规映射	✓	P5：不属于管理员范围
98	A3-管理	M10-检测规则	✓	P5：不属于管理员范围
99	A3-管理	M11-AAR档案	✓	P5：不属于管理员范围
100	A3-管理	M12-BCP/风险登记册	×	P4+P5：MMA介导、只读
A4（报告） →记忆				
101	A4-报告	M1-威胁存储库	✓	P5：报告范围之外
102	A4-报告	M2-资产清单	✓	P5：报告范围之外
103	A4-报告	M3-策略存储库	✓	P5：报告范围之外
104	A4-报告	M4-SIEM数据湖	✓	P5：报告范围之外
105	A4-报告	M5-代码仓库	✓	P5：报告范围之外
106	A4-报告	M6-案例管理	✓	P5：报告范围之外
107	A4-报告	M7-CTI知识库	✓	P5：报告范围之外
108	A4-报告	M8-剧本仓库	×	P4+P5：MMA介导、改进循环
109	A4-报告	M9-合规映射	×	P4+P5：MMA介导、改进循环
110	A4-报告	M10-检测规则	×	P4+P5：MMA介导的改进循环
111	A4-报告	M11-AAR档案	×	P4+P5：MMA介导的改进循环
112	A4-报告	M12-BCP/风险登记册	✓	P5：报告范围之外

类别2摘要： 16 保留（×），32 消除（✓）。缩减：67%　　.

类别 3：智能体 ↔智能体边界 共 *12* 项，保留 *4* 项，消除 *8* 项（*67%*）

In the flat baseline，每个智能体 pair 均进行双向通信（6 对 × 2 方向 = 12）。AgenticCyOps 将代理间通信限制为按顺序进行的主机介导阶段交接。仅保留四项个单向交接，每项均将主机作为中间信任检查点。

#	源	目的地	状态	机制
113	A1-监控	A2-分析	×	主机介导，P1+P3
114	A2-分析	A1-监控	✓	无反向交接
115	A1-监控	A3-管理	✓	非相邻阶段
116	A3-管理	A1-监控	✓	非相邻阶段
117	A1-监控	A4-报告	✓	非相邻阶段
118	A4-报告	A1-监控	×	宿主介导的反馈回路
119	A2-分析	A3-管理	×	主机介导，P1+P3
120	A3-管理	A2-分析	✓	无反向交接
121	A2-分析	A4-报告	✓	非相邻阶段
122	A4-报告	A2-分析	✓	非相邻阶段
123	A3-管理	A4-报告	×	主机介导，P1+P3
124	A4-报告	A3-管理	✓	无反向交接

类别 3 摘要：保留 4 项（×），消除 8 项（✓）。缩减：67%。

类别 4：工具响应 →智能体边界总计 *64* 项，保留 *16* 项，消除 *48* 项（*75%*）工具响应被视为独立边界，因为返回载荷可能携带独立于出站请求授权的注入内容。在扁平基线中，任何工具都可以向任何智能体返回数据。AgenticCyOps 将响应限制为请求智能体在相应阶段范围内使用的工具。保留的边界须通过签名清单（P1）进行响应验证，并通过共识验证（P3）。

#	Tool	智能体	状态	机制
监控阶段工具 *(T1–T4)* → 代理 *[4工具]*				
125	T1-UEBA 模型	A1-监控	×	P1+P3：已签名响应，验证循环
126	T1-UEBA 模型	A2-分析	✓	P2：响应在主机处被阻止
127	T1-UEBA 模型	A3-管理	✓	P2：响应在主机处被阻止
128	T1-UEBA 模型	A4-报告	✓	P2：响应在主机处被阻止
129	T2-入侵检测系统/入侵防御系统/配置管理数据库	A1-监控	×	P1+P3：已签名响应，验证循环
130	T2-入侵检测系统/入侵防御系统/配置管理数据库	A2-分析	✓	P2：响应在主机处被阻止
131	T2-入侵检测系统/入侵防御系统/配置管理数据库	A3-管理员	✓	P2：响应在主机处被阻止
132	T2-入侵检测系统/入侵防御系统/配置管理数据库	A4-报告	✓	P2：响应在主机处被阻止
133	T3-EDR/NDR 传感器	A1-监控	×	P1+P3：已签名响应，验证循环
134	T3-EDR/NDR 传感器	A2-分析	✓	P2：响应在主机处被阻止
135	T3-EDR/NDR 传感器	A3-管理员	✓	P2：响应在主机处被阻止
136	T3-EDR/NDR 传感器	A4-报告	✓	P2：响应在主机处被阻止
137	T4-ITSM/工单管理	A1-监控	×	P1+P3：已签名响应，验证循环
138	T4-ITSM/工单管理	A2-分析	✓	P2：响应在主机处被阻止
139	T4-ITSM/工单管理	A3-管理	✓	P2：响应在主机处被阻止
140	T4-ITSM/工单管理	A4-报告	✓	P2：响应在主机处被阻止
分析阶段工具（*T5–T7*）→代理 *[3 工具]*				
141	T5-沙箱	A1-监控	✓	P2：响应在主机处被阻止
142	T5-沙箱	A2-分析	×	P1+P3：已签名响应，RCA 循环
143	T5-沙箱	A3-管理	✓	P2：响应在主机处被阻止
144	T5-沙箱	A4-报告	✓	P2：响应在主机处被阻止
145	T6-SIEM 搜索	A1-监控	✓	P2：响应在主机处被阻止
146	T6-SIEM 搜索	A2-分析	×	P1+P3：已签名响应，RCA 循环
147	T6-SIEM 搜索	A3-管理	✓	P2：响应在主机处被阻止
148	T6-SIEM 搜索	A4-报告	✓	P2：响应在主机处被阻止
149	T7-代码分析器	A1-监控	✓	P2：响应在主机处被阻止
150	T7-代码分析器	A2-分析	×	P1+P3：已签名响应，RCA 循环
151	T7-代码分析器	A3-管理	✓	P2：响应在主机处被阻止
152	T7-代码分析器	A4-报告	✓	P2：响应在主机处被阻止
管理阶段工具（*T8–T12*）→ 代理 *[5工具]*				
153	T8-IAM/PAM	A1-监控	✓	P2：响应在主机处被阻止
154	T8-IAM/PAM	A2-分析	✓	P2：响应在主机处被阻止
155	T8-IAM/PAM	A3-管理	×	P1+P3：已签名响应，恢复循环
156	T8-IAM/PAM	A4-报告	✓	P2：响应在主机处被阻止
157	T9-防火墙/APM	A1-监控	✓	P2：响应在主机处被阻止
158	T9-防火墙/APM	A2-分析	✓	P2：响应在主机处被阻止
159	T9-防火墙/APM	A3-管理	×	P1+P3：已签名响应，恢复循环
160	T9-防火墙/APM	A4-报告	✓	P2：响应在主机处被阻止
161	T10-配置管理器	A1-监控	✓	P2：响应在主机处被阻止
162	T10-配置管理器	A2-分析	✓	P2：响应在主机处被阻止
163	T10-配置管理器	A3-管理	×	P1+P3：已签名响应，恢复循环
164	T10-配置管理器	A4-报告	✓	P2：响应在主机处被阻止
165	T11-EPP/AV	A1-监控	✓	P2：响应在主机处被阻止
166	T11-EPP/AV	A2-分析	✓	P2：响应在主机处被阻止
167	T11-EPP/AV	A3-管理	×	P1+P3：已签名响应，恢复循环
168	T11-EPP/AV	A4-报告	✓	P2：响应在主机处被阻止
169	T12-ANSIBLE	A1-监控	✓	P2：响应在主机处被阻止
170	T12-ANSIBLE	A2-分析	✓	P2：响应在主机处被阻止
171	T12-ANSIBLE	A3-管理	×	P1+P3：已签名响应，恢复循环

续见下一页

#	Tool	智能体	状态	机制
172	T12-ANSIBLE	A4-报告	✓	P2：响应在主机处被阻止
报告阶段工具 *(T13–T16)* → 代理 *[4工具]*				
173	T13-报告仪表板	A1-监控	✓	P2：响应在主机处被阻止
174	T13-报告仪表板	A2-分析	✓	P2：响应在主机处被阻止
175	T13-报告仪表板	A3-管理	✓	P2：响应在主机处被阻止
176	T13-报告仪表板	A4-报告	×	P1+P3：已签名响应，改进循环
177	T14-ISAC/MISP	A1-监控	✓	P2：响应在主机处被阻止
178	T14-ISAC/MISP	A2-分析	✓	P2：响应在主机处被阻止
179	T14-ISAC/MISP	A3-管理	✓	P2：响应在主机处被阻止
180	T14-ISAC/MISP	A4-报告	×	P1+P3：已签名响应，改进循环
181	T15-编辑器与测试	A1-监控	✓	P2：响应在主机处被阻止
182	T15-编辑器与测试	A2-分析	✓	P2：响应在主机处被阻止
183	T15-编辑器与测试	A3-管理	✓	P2：响应在主机处被阻止
184	T15-编辑器与测试	A4-报告	×	P1+P3：已签名响应，改进循环
185	T16-GRC 映射器	A1-监控	✓	P2：响应在主机处被阻止
186	T16-GRC 映射器	A2-分析	✓	P2：响应在主机处被阻止
187	T16-GRC 映射器	A3-管理	✓	P2：响应在主机处被阻止
188	T16-GRC 映射器	A4-报告	×	P1+P3：已签名响应，改进循环

类别4摘要： 保留 16 项（×），消除 48 项（✓）。缩减：75%　　.

类别5：外部情报源 → 内存边界　　共 *12* 项, 保留 *4* 项, *8* 项已消除 *(67%)*

In the flat baseli[ne, eac] h external feed writes directly t

直接写入相应的记忆存储库，未经任何中介。AgenticCyOps 将所有外部数据摄取都经由记忆管理智能体（MMA）进行，该智能体会应用写入边界过滤（P4），并将数据源限制为指定的存储库（P5）。仅保留四条从情报源到存储库的路径。

#	外部情报源	记忆存储	状态	机制
189	E1-供应商公告	M7-CTI知识库	×	P4+P5：MMA 写入过滤
190	E2-信息共享与分析中心情报源	M7-CTI知识库	✓	P5：通过E1路径汇总
191	E3-MISP 社区	M1-威胁存储库	×	P4+P5：MMA 写入过滤
192	E4-CVE/NVD	M1-威胁存储库	✓	P5：通过E3路径整合
193	E5-OSINT 来源	M7-CTI知识库	✓	P5：通过E1路径整合
194	E6-法规更新	M9-合规映射	×	P4+P5：MMA 写入过滤
195	E7-威胁情报订阅	M1-威胁存储库	✓	P5：通过 E3路径汇总
196	E8-漏洞扫描器	M4-SIEM数据湖	✓	P5：通过监控工具摄取
197	E9-合作伙伴安全运营中心信息源	M7-CTI知识库	✓	P5：通过E1路径汇总
198	E10-行业基准	M9-合规映射	✓	P5：通过 E6 路径整合
199	E11-补丁公告	M10-检测规则	×	P4+P5：MMA 写入过滤
200	E12-合规更新	M9-合规映射	✓	P5：通过 E6 路径整合

类别 5 摘要： 4 项保留（×），8 项消除（✓）。缩减：67%。

信任边界验证摘要

类别	Flat	AgenticCyOps	消除
类别 1：智能体 → 工具（#1–64）	64	16	48
类别 2：智能体 → 记忆（#65–112）	48	16	32
类别 3：智能体 ↔ 智能体（#113–124）	12	4	8
类别 4：工具响应 → 智能体（#125–188）	64	16	48
类别 5：外部信息源 →记忆（#189–200）	12	4	8
总计	**200**	**56**	**144**

缩减：144/200 = 72%。 所有 56 个保留边界都至少受到一种主动验证机制（P1、P3、P4 或 P5）的约束。