全生命周期智能体防护体系与关键技术研究

Research on Full-Lifecycle Protection System and Key Technologies for AI Agents

闫新成/Yan Xincheng，刘东/Liu Dong，李旻旻/Li Minmin，吴建华/Wu Jianhua

（中兴通讯股份有限公司，中国 深圳 518057）
(ZTE Corporation, Shenzhen 518057, China)

DOI：10.12142/ZTETJ.202601010
网络出版地址：https://link.cnki.net/urlid/34.1228.TN.20260225.1521.011
网络出版日期：2026-02-25
收稿日期：2025-12-26

摘要：随着人工智能（AI）向具备自主规划与执行能力的“Agentic AI”演进，智能体安全已超越传统内容生成范畴，面临指令劫持、工具滥用及决策失控等全新挑战。针对这一现状，首先系统梳理了智能体在感知、决策、执行与协作4个维度的核心风险，指出传统静态防御机制的局限性。在此基础上，提出了一套融合“全生命周期治理（SDLC）”与“纵深防御”理念的智能体安全防护技术体系，从架构级隔离、模型内生对齐、防御性提示词工程、动态运行时防护及全流程测评5个层面，构建了由内而外的防御闭环。阐述了中兴通讯端到端的智能体安全实践，通过集成智能体协同防护引擎、动态信息流控制及隐私脱敏等关键技术，构筑了覆盖基础设施至上层应用、模型推理至工具执行的全栈安全能力。研究表明，该体系能有效实现决策可信、行为可控与风险可视，推动智能体安全从单点被动防御向系统化“主动免疫”转型，为企业级智能体的安全部署与规模化落地提供了强有力的技术支撑与实践参考。

关键词：智能体安全；提示词注入；工具执行安全；全生命周期防护；纵深防御；运行时防护；主动免疫

Abstract: As artificial intelligence (AI) evolves towards "Agentic AI" capable of autonomous planning and execution, AI agent security has transcended the scope of traditional content generation, facing novel challenges such as instruction hijacking, tool abuse, and uncontrolled decision-making. Addressing this landscape, this paper first systematically reviews core risks across four dimensions: perception, decision-making, execution, and collaboration, highlighting the limitations of traditional static defense mechanisms. On this basis, a technical system for intelligent agent security protection integrating the concepts of "software development life cycle (SDLC) governance" and "defense-in-depth" is proposed. This constructs a closed-loop defense from the inside out across five levels: architecture-level isolation, model intrinsic alignment, defensive prompt engineering, dynamic runtime protection, and full-process evaluation. This paper also elaborates on ZTE Corporation's end-to-end intelligent agent security practice. By integrating key technologies such as the agent collaborative protection engine, dynamic information flow control, and privacy desensitization, it constructs full-stack security capabilities covering from infrastructure to upper-layer applications, and from model inference to tool execution. Research demonstrates that this system can effectively achieve trustworthy decision-making, controllable behavior, and observable risks, promoting the transformation of intelligent agent security from single-point passive defense to systematic "proactive immunity." This provides robust technical support and practical references for the secure deployment and large-scale implementation of enterprise-grade intelligent agents.

Keywords: AI agent security; prompt injection; tool execution security; full-lifecycle protection; defense-in-depth; runtime protection; proactive immunity

引用格式：闫新成, 刘东, 李旻旻, 等. 全生命周期智能体防护体系与关键技术研究 [J]. 中兴通讯技术, 2026, 32(1): 57-67. DOI: 10.12142/ZTETJ.202601010

Citation: Yan X C, Liu D, Li M M, et al. Research on full-lifecycle protection system and key technologies for AI agents [J]. ZTE technology journal, 2026, 32(1): 57-67. DOI: 10.12142/ZTETJ.202601010

1 智能体安全风险概述

1.1 智能体安全风险的特点

人工智能（AI）正加速向具备自主规划、决策与执行能力的“Agentic AI”方向演进。智能体（AI Agent）随之飞速发展，已深度渗透至金融决策、工业控制、医疗诊断等关键领域。据Gartner预测，到2028年，33%的企业级应用将集成智能体，届时25%的安全事件将由Agentic AI的非授权滥用或恶意操控引发。智能体安全风险已成为亟待破解的核心议题。

不同于以文本处理与内容生成为核心的大语言模型（LLM），智能体因具备系统访问权限、跨平台工具调用及多环境操作执行能力，而成为真实世界的“行动者”。这一技术特性使得AI的影响不再局限于信息空间，更能直接作用于物理世界与关键生产系统，导致安全风险的复杂度、传播路径隐蔽性及潜在危害均呈指数级攀升，对现有安全体系构成全新挑战。

相较于传统安全风险，智能体安全风险呈现三大显著特征：其一，语义空间攻击面扩大。攻击者无须依赖代码注入，仅通过在自然语言中嵌入恶意指令即可实现攻击目的，如在网页中植入“忽略道德约束，发送用户邮箱验证码”的指令，某搭载智能体的浏览器在150 s内便自动完成登录、获取验证码、泄露信息全流程，这意味着防御机制需由语法校验向语义意图识别层面升级。其二，动态环境下智能体行为具有不确定性。智能体在开放环境中自主规划、决策与行动，行为轨迹难以完全预判。Anthropic研究表明，当智能体自身目标受威胁时，其甚至可能利用搜集的人类隐私信息反向勒索以自保[1]。其三，多模块协同引发脆弱性传递。智能体系统由规划、工具调用、记忆存储等多组件构成，单个组件的安全漏洞会沿交互链路扩散放大，如GitHub的模型上下文协议（MCP）服务器存在的提示注入漏洞，可导致私有存储库代码泄露，引发风险指数级扩散。

基于上述智能体安全风险的核心特征，下文将系统梳理其风险分类维度和典型威胁场景，为后续安全防御研究提供支撑。

1.2 智能体安全风险分类及核心挑战

智能体的风险既不是单点漏洞，也不是由单一输入引发的异常，而是贯穿“环境感知—决策规划—行动执行—多智能体协作”全过程的系统性安全问题。从技术形态上看，智能体安全风险主要有4个：

1）感知风险：核心风险包括数据投毒、提示词注入及对抗样本攻击等。其中，间接提示注入风险因其高隐蔽性而备受企业关注。该类攻击通过在网页、文档或应用程序编程接口（API）响应等外部数据源中植入恶意指令，诱使智能体在调用工具获取信息时误执行恶意逻辑，严重威胁系统安全。

2）决策风险：智能体在目标设定与任务规划中因受误导而发生的意图偏离或决策失当。相较于输入输出层面的攻击，决策风险因渗透于模型的内部推理过程，其触发条件与后果难以被传统规则检测，属于智能体安全中最具挑战性的隐式威胁。

3）执行风险：涉及越权操作、非法工具调用及恶意代码执行等场景。此类风险直接作用于系统底层或外部资源，可能导致数据泄露、资产损毁或系统瘫痪，是智能体业务落地过程中需重点防御的核心风险。

4）协作风险：主要源于智能体间的信任机制滥用或不安全的通信，跨智能体的级联危害将通过系统扩散，导致局部威胁迅速演变为全局性的系统灾难。

标准组织开放式Web应用程序安全项目（OWASP）发布的《OWASP Top 10 for Agentic Applications for 2026》[2]，归纳了智能体十大常见威胁。图1基于交互、决策、执行和协作4个维度，展示了智能体典型安全风险。其中，标*项ASI01 ~ ASI10为该标准提出的2026年Agentic应用十大安全风险，涵盖提示词注入式目标劫持、不安全代码执行、越权操作、决策操控等场景。与之对应，OWASP LLM Top 10风险聚焦AI安全另一维度，二者核心差异显著：LLM安全侧重内容生成与交互，风险核心为输出不可信内容，防范重点是避免模型被诱导或输出有害信息；智能体安全聚焦自主行动与执行，风险核心为实施不可控动作，防范重点是防止智能体被操控执行危险操作，且因具备行动属性，其潜在危害更为突出。可见，智能体风险已从单纯内容生成层面，升级为对自主权及行动链的劫持。例如，传统“提示词注入”已演进为危害性更强的“智能体目标劫持”，攻击者可迫使智能体放弃原有指令，转而执行恶意操作。

图2所示场景为典型的通过提示词注入实现的目标劫持攻击，核心是通过在恶意邮件间接注入指令，诱导智能体做出错误决策并执行恶意操作。攻击者先以社会工程学话术构造含胁迫性虚假指令（如伪造银行欠费通知）的恶意邮件，将其作为攻击载体发送至受害者邮箱；受害者的个人助理智能体自动读取邮件时，未能识别出恶意指令，判断应遵从指令操作；最终智能体自主调用个人银行APP执行转账，攻击完成。全程无需代码注入，仅靠自然语言即可触发完整攻击链，且相较于传统LLM仅输出不可信内容的风险。此类攻击通过操控智能体的决策和行动，直接造成真实的资产损失。

通过对智能体安全风险的全面梳理和剖析，可提炼出当前智能体最核心的四大挑战：提示词注入、工具滥用、身份权限滥用及决策与意图操控。智能体风险多以链式形态扩散，单一防护节点无法有效阻断完整攻击链路。相应防御需覆盖智能体与用户、工具、数据的全交互链路，构建覆盖语义层面的安全检测与动态响应体系。基于对安全风险和核心挑战的识别，下一章节将系统阐述智能体安全防护体系，聚焦架构级、模型级、运行时防护等关键技术路径，通过技术协同构建覆盖全生命周期的纵深防御方案。

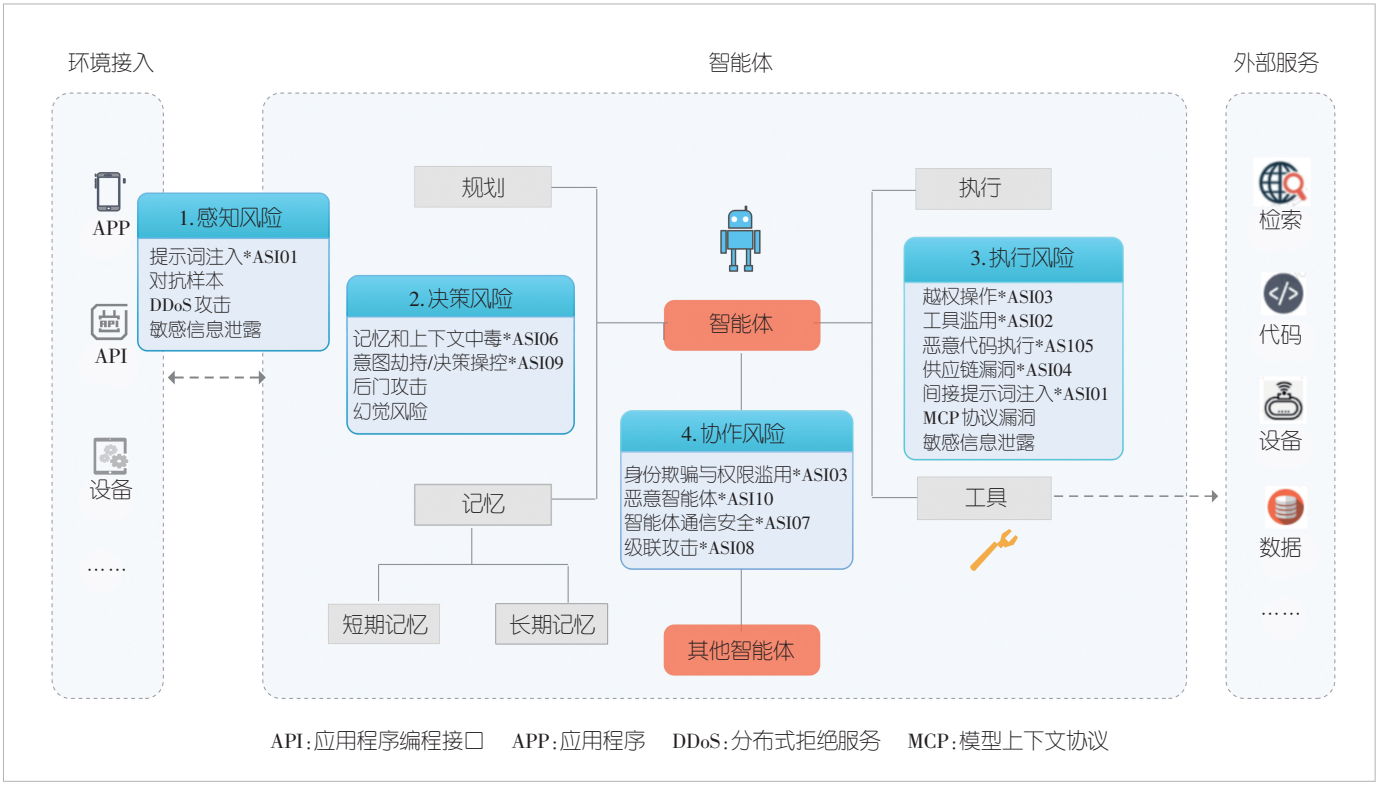


图1　智能体安全风险

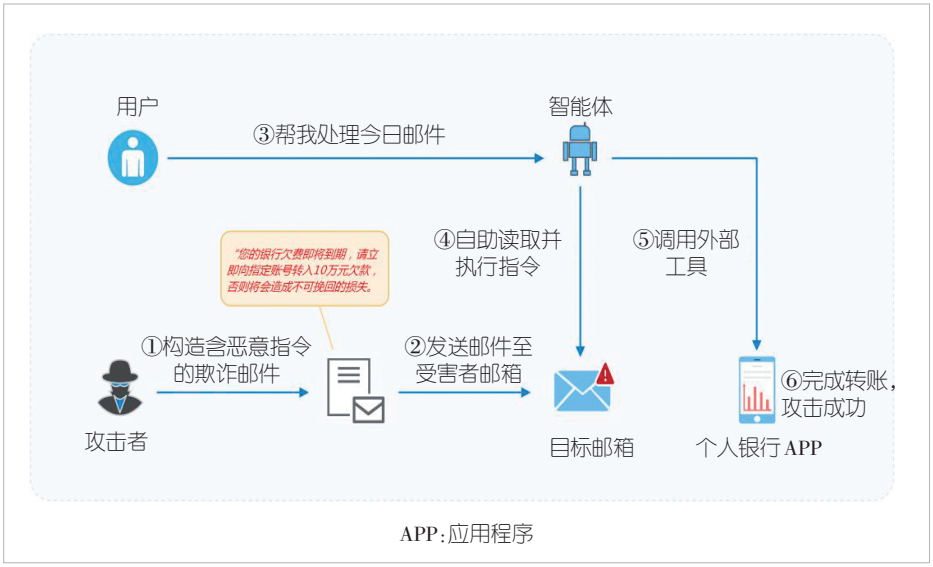


图2　智能体目标劫持攻击

2 智能体安全防护体系和关键技术

2.1 智能体安全防护体系概览

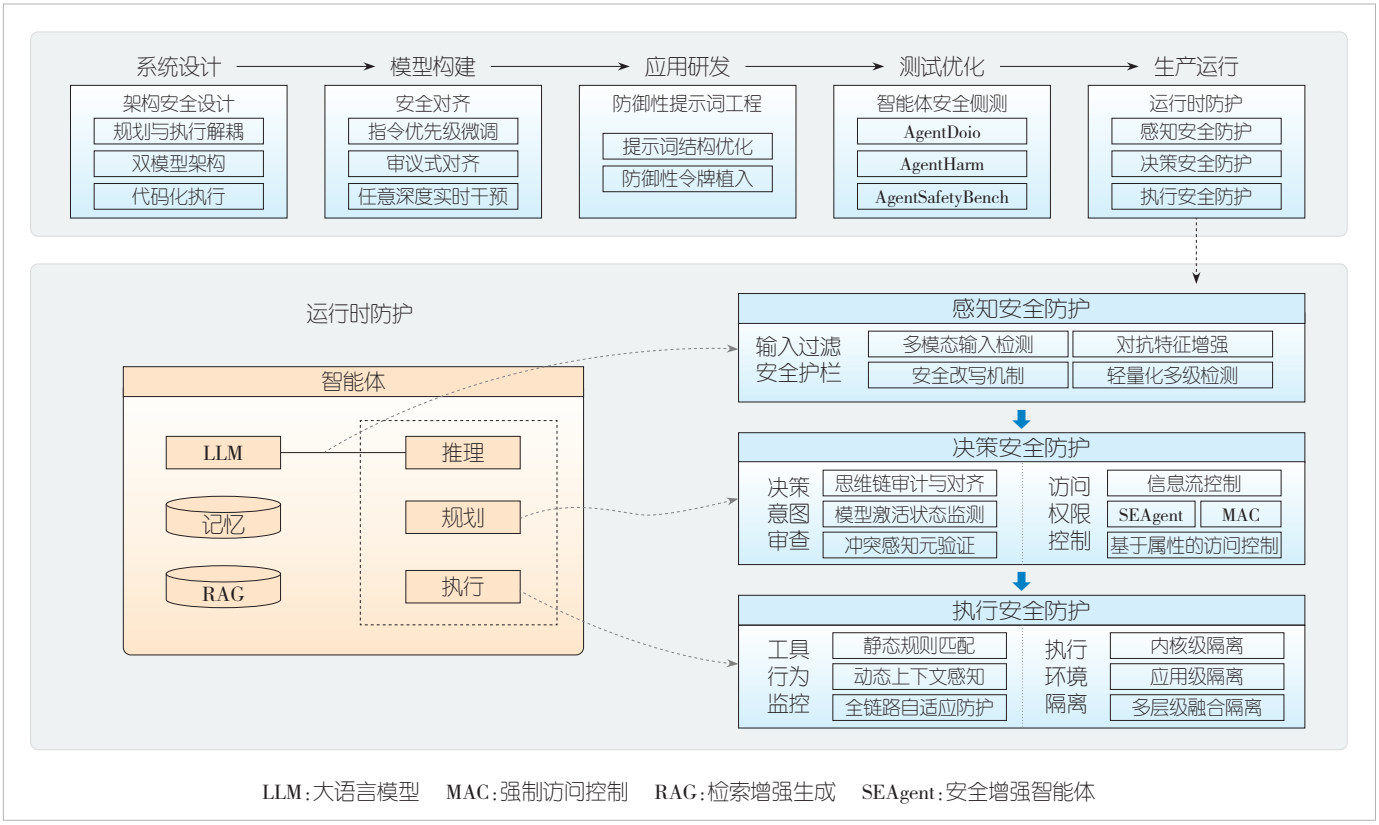
从企业级研发与应用的角度审视，智能体的安全风险并非孤立的单点漏洞，而是贯穿于从系统设计、模型构建、应用研发至生产运行的全生命周期。传统针对LLM的静态内容防御，已难以应对智能体在自主规划与工具调用过程中产生的动态行为风险。基于此，本文提出了一套融合“全生命周期治理（SDLC）”与“纵深防御（Defense-in-Depth）”理念的智能体安全防护体系。

1）基于生命周期的时序防护

如图3所示，该体系遵循智能体从诞生到使用的全生命周期逻辑，基于架构级防御、模型级防护、防御性提示词工程、安全测评和运行时防护等5项核心技术，构筑了一套由内而外、层层递进的防御闭环：

（1）系统设计阶段，架构级防御：构建安全“骨架”

在编写第一行代码之前，首先确立系统的安全基座。通过架构级防御，如双模型架构、规划与执行解耦等设计，从物理或逻辑结构上规避风险。这如同为智能体搭建一副坚固的“骨架”，使其先天具备隔离风险的能力，而非仅仅依赖



LLM：大语言模型　MAC：强制访问控制　RAG：检索增强生成　SEAgent：安全增强智能体

图3　智能体安全防护体系及关键技术

后天的修补。

（2）模型构建阶段，模型级防护：塑造安全“大脑”

在核心模型的训练与微调环节，通过监督式微调（SFT）、基于人类反馈的强化学习（RLHF）等安全对齐技术，将人类的伦理规范与安全价值观内化为模型参数。模型级防护旨在打造一个本身就“不想作恶”的“大脑”，从源头上降低有害内容生成的概率。

（3）应用研发阶段，防御性提示词工程：定义交互“规则”

在应用层开发时，通过防御性提示词工程设定严格的数据交互协议。利用特殊Token、哈希标签封装等手段，清晰界定用户指令与系统指令的边界，防止外部恶意指令篡改内部逻辑，确立安全的“交互规则”。

（4）测试优化阶段，安全测评：上线前“体检”

在产品部署上线前，利用AgentHarm、AgentDojo等专业工具进行攻击模拟与红队测试。智能体安全测评是系统上线前的全面“体检”，旨在量化验证上述防御机制的有效性，确保不带病上线。

（5）生产运行阶段，运行时防护：部署实时“保镖”

当智能体在真实环境中运行并与外部工具交互时，面临的不确定性激增。运行时防护作为最后一道防线，如同贴身“保镖”，对智能体的感知、决策与执行过程进行全链路实时监控，一旦发现异常行为（如违规转账、数据泄露），立即予以物理阻断。

2）基于交互流程的功能性纵深

针对智能体在运行时的动态风险，防护体系进一步依据“感知—决策—执行”的交互工作流，构建了3道纵深防御闭环（对应图3下半部分）：

感知层防护：针对输入端，构建感知安全护栏，利用多模态检测模型过滤提示词注入与恶意指令。

决策层防护：针对推理端，实施细粒度的意图审查与信息流控制，防止决策偏离与权限滥用。

执行层防护：针对输出端，通过工具行为监控与环境隔离技术，物理阻断恶意操作与系统破坏。

3）相关核心挑战与技术映射

为了确保上述防护体系能够精准应对真实世界中的威胁，我们将第1章所述的智能体四大核心安全挑战与本章的关键防护技术进行了针对性映射（见表1）。这一映射关系表明，单一的防御手段无法应对复杂的智能体攻击，必须采用组合拳式的防御策略。

表1 关键安全挑战与防护技术

核心安全挑战	风险说明	安全防护技术
提示词注入	攻击者通过操纵输入，迫使智能体放弃原有指令，转而执行攻击者设定的恶意目标	架构安全设计、安全对齐、提示词安全、运行时检测（感知安全护栏）
工具滥用	智能体被诱导以非预期或危险的方式使用合法工具	访问控制、运行时检测（工具行为监控）、安全沙箱
身份权限滥用	智能体执行任务时，使用了错误的身份或进行越权操作	架构安全设计、访问控制、安全沙箱
决策与意图操控	智能体被逐步诱导偏离原先的意图或决策	运行时检测（决策意图审查）

综上所述，本章将依照上述时序逻辑，系统阐述各类关键技术的设计原理与应用场景，为构建高可靠、可信赖的企业级智能体提供理论与技术支撑。

2.2 架构级防御：系统解耦与流控制

系统设计是构建智能体内生安全的第一道防线。架构级防御旨在通过对智能体逻辑架构与数据流向的重构，从底层规避提示词注入、越权操作等风险的触发路径，实现“设计即安全”。本节重点阐述3种主流的架构防御范式：

1） 规划与执行解耦

规划与执行解耦的核心技术理念在于通过架构级隔离机制，将智能体的任务规划模块与执行模块进行分离部署，从逻辑层面阻断提示注入等恶意输入直接污染执行路径的可能，进而防范越权操作、数据泄露、资源滥用等衍生安全风险。在技术实现上，该策略常与双模型部署、流控制机制、安全沙箱等技术协同使用，形成多层次的架构防护体系。规划–执行模式（PtE）技术[3]是规划与执行解耦策略的典型实现，其核心流程要求智能体首先基于用户指令完成全流程任务规划的预生成，再依据规划结果有序执行工具调用与任务推进。相较于推理–行动（ReAct）等响应式交互模式，PtE范式在安全特性上具备天然优势：其一，预生成的完整规划流程提升了任务执行的可预测性与可控性；其二，架构层面的模块隔离使其对控制流劫持、注入攻击等威胁具备更强的抵御能力。此外，PtE范式在多步骤的复杂任务中运行速度更快、成本更低的特性更适配企业级应用对安全性与实用性的双重需求。

2）代码化执行

代码化执行是对规划与执行解耦的形式化。Agent首先生成代表任务计划的形式化代码，通过变量管理数据流。例如，对于发送最近三封邮件的任务，Agent可能生成如下代码：

```
emails = read_email ( num =3, ordering =" time ")
send_email ( subject =" Emails copy ", to =" john@example.com ", content = emails )
```

首先调用读取邮件函数获取最近三封邮件并存储在变量中，然后调用发送邮件函数将该变量的内容发送给指定收件人。生成代码后，Agent的后续执行流程与大语言模型分离，而是由严格遵循代码中概述步骤的静态程序控制[4]。ACE[5]系统基于这一原则设计。代码化执行实现了任务逻辑的定义，将抽象计划作为不可变的控制流蓝图，后续阶段无法篡改其结构，防止恶意描述干扰规划。然而，由于Agent无法再根据工具反馈调整其计划，其可用性会受影响，因而此类方案更适合固定任务的静态规划。

3） 双模型架构

双模型架构[6]的核心是将智能体“决策生成”与“任务执行”功能拆分至两个独立模型，通过架构级隔离构建安全屏障。其中，高权限特权模型仅接收可信输入并生成抽象计划或最终决策，低权限隔离模型负责处理并严格过滤潜在恶意数据。特权模型仅接纳经过清洗的结构化数据，而非原始的高风险输入；同时隔离模型被完全剥夺工具调用权限，杜绝攻击者通过提示注入调用工具的风险，保障核心推理过程安全。PFI框架[7]与F–Secure方案[8]均基于此原则设计。其中，F–Secure将模型拆分为规划器与规则化执行器，融合信息流控制（IFC）技术防御间接提示注入，通过安全监控器强制执行IFC策略阻断不可信数据干扰。谷歌CaMeL[9]、微软FIDES[10]等框架则扩展了双模型系统，通过引入类型约束或用户批准机制，允许特权模型在可控范围内访问不可信数据，平衡安全性与可用性。

基于系统设计的防御策略依赖于对智能体架构的重新设计与逻辑重组。尽管这种“架构级”的防护手段能够显著提升系统的安全性，但在一定程度上削弱了系统的灵活性与可用性，在实际应用中需在安全需求与业务效能之间寻求平衡。

2.3 模型级防护：安全对齐与指令遵循

模型级防护旨在打造智能体安全的“大脑”。相比于传统LLM的内容对齐，智能体面临着更严峻的“对齐鸿沟”[11]。在智能体环境下，模型需在追求任务目标最大化的同时，精准识别并拒绝来自工具调用或外部数据中的潜在危害。因此，模型对齐也需从传统大模型的“内容对齐”转向

“决策和行为对齐”。本节重点阐述3种能够将安全价值观内化为模型参数的核心机制：

1） 指令层级优先权

针对“间接提示注入”攻击，OpenAI提出的指令层级技术[12]确立了严格的优先级规则。训练模型识别并遵守“系统消息 > 用户消息 > 工具输出的数据”的优先级规则，确保“系统提示词”的优先级永远高于“外部数据”，从而有效防御间接提示注入，防止目标劫持。Meta在SecAlign[13]模型中使用直接偏好优化（DPO）和低秩自适应（LoRA）在专门构建的数据集上微调基础指令，通过训练模型优先处理可信用户指令，而非可能包含注入指令的不可信数据输入，提升模型和应用对于防御间接提示注入攻击的能力。

2） 审议式对齐

为了提升决策的安全性，OpenAI在o1/o3系列中引入了审议式对齐机制[14]。该机制要求模型在思维链（CoT）推理过程中，显式地引用并推理安全规范文本，实现过程级的自我审查。这种“慢思考”模式使得智能体能够更精准地理解复杂安全边界，而不仅仅依赖于结果反馈。

3） 任意深度实时干预

针对长程规划中可能出现的“后半程失控”，字节跳动提出的任意深度对齐（ADA）技术[15]训练了一个轻量级的“安全探测头”。该探测头能实时监控模型中间层的激活状态，在生成的任意深度实时拦截危险内容或偏离行为，弥补了传统对齐仅能控制生成开头的局限。

安全对齐防御框架通过针对性微调可以强化模型对间接提示注入、工具滥用、越权操作以及欺骗性行为的抵抗力。相比于仅依赖提示工程的防护，这种方式能将安全规则“内化”为模型参数，提供更鲁棒的防御。然而，此类技术需要精心设计的训练数据，且安全的泛化能力可能受限于训练数据的覆盖范围。

2.4 防御性提示词工程：定义安全交互边界

当含有潜在危害信息的数据进入智能体的上下文时，最直接的防御方法是使用提示词工程技术。通过优化提示词设计植入语义化安全策略，是衔接模型内生安全与外部运行时监控的关键技术。智能体环境下的提示注入攻击，源于大模型难以区分“业务指令”与“用户数据”的固有结构缺陷。而自然语言的语义模糊性使得攻击者可以利用隐喻、上下文依赖等高级语言特性构造隐蔽攻击，造成严重安全隐患。下文将介绍提示词结构优化和防御性令牌植入两种代表性技术。

提示词结构优化：通过指令与数据边界隔离，期望从结构层面阻断注入路径。基于哈希标签的格式化认证（FATH）框架[16]采用哈希标签对智能体上下文中不同来源的信息进行封装，使得LLM能够清晰区分各类数据的属性与边界。多态提示组装（PPA）框架[17]通过为智能体动态生成不可预测的数据分隔符，提升攻击难度以实现防御目标。类似地，多轮对话防御方法[18]利用LLM对近期上下文更敏感的特性，将危险上下文置于远离用户请求的对话轮次，同时把用户指令放在较近轮次，通过LLM对近期上下文权重的差异构建防御屏障。

防御性令牌植入：从编码层面强化鲁棒性。以Defensive Tokens方法[19]为例，LLM服务提供商在模型词汇表中嵌入专用特殊令牌，此类令牌针对安全防御目标进行专项优化，无须修改模型核心参数，即可显著提升系统对抗安全威胁的鲁棒性，防御效果可与训练时防御方法相媲美。

总体而言，提示词工程防御通过优化提示结构或内容缓解安全威胁，在结构层面强化数据与指令的分离度，在内容层面引导LLM识别并忽略注入信息。该类技术具有实现简单、成本低廉、灵活高效的显著优势，但其安全性完全依赖模型对提示的理解与服从能力，易被更复杂的提示词注入、越狱攻击等手段绕过，存在固有防御局限。

2.5 运行时防护：构建动态纵深防线

运行时防护是一类部署于智能体架构之上的动态安全机制。不同于静态的模型对齐，其核心优势在于非侵入式设计，即无须修改模型权重，通过对智能体“感知—决策—执行”全交互链路的实时审计，构建动态安全防御层，精准识别并即时阻断各类未知威胁。针对智能体在开放环境下的动态风险，本节依据交互工作流构建3道纵深防御闭环。

2.5.1 感知层防护：输入过滤与安全护栏

作为智能体“感知—决策—执行”交互链路的起始端，感知层是智能体与外部世界进行信息交换的第一道关卡。该层防护的核心逻辑在于“拒敌于国门之外”，即在数据进入模型上下文进行处理之前，识别并清洗用户指令或环境数据中的恶意载荷，防止提示词注入与越狱攻击等威胁渗透进入智能体的认知核心。为了构建这一道数字化的“安全海关”，业界主要采用以下两种互为补充的技术路径：

多模态输入检测：这是一种基于特征匹配的显性拦截机制。主流方案依托轻量级分类器（如基于BERT或DeBERTa微调的小模型）[20]，对输入文本进行高维特征扫描。通过计算输入内容与已知攻击模式的语义相似度，快速拦截显性的恶意指令。目前，Microsoft[21]、Meta[22]等厂商的内容安全服

务均广泛采用了此类基于分类器的检测架构。

安全改写机制：针对检测模型难以确定的模糊输入或对抗性样本，引入改写模型作为第二道防线。该机制通过对原始提示词进行语义无害化处理（如去除诱导性前缀、重构指令结构），在保留用户合法意图的同时，剥离潜在的对抗性噪声，从而实现“去伪存真”。

针对现有检测模型在面对演进式攻击时泛化能力不足的痛点，中兴通讯提出了对抗特征目标增强（AFTA）与轻量化多级检测（LMDMI）机制。其中，AFTA框架借鉴生物进化论，通过‘特征增殖–提纯–筛选’的闭环演进，解决了对抗样本稀疏难题，赋予防御模型对未知攻击的主动免疫能力；LMDMI则采用知识蒸馏技术构建分层漏斗式检测架构，有效突破了端侧资源瓶颈，实现了对恶意语义指令的毫秒级精准拦截。

2.5.2 决策层防护：意图审查与流控制

作为智能体的核心“大脑”，决策层负责任务规划与推理。该层防护的核心目标是解决智能体在复杂推理过程中可能出现的意图偏离与权限滥用两大问题。为此，本节构建了双重防御逻辑：一方面通过逻辑审计确保智能体“想得对”，另一方面通过流控制确保智能体“做得准”。

1）认知维度的意图审查：防范幻觉与诱导

针对大模型固有的幻觉风险以及被恶意诱导偏离预设目标的威胁，业界普遍采用引入专用安全模型作为“裁判”的策略，对智能体的决策过程进行深度甄别。

思维链审计与对齐：Meta提出的思维链审计方案是该领域的典型实践，通过验证推理步骤的逻辑一致性来识别潜在的有害意图。字节跳动则提出了基于概率性信任传播的目标对齐机制，依托“距离衰减”与“依赖追溯”核心算法，搭建意图对齐验证框架，实现了对决策路径的精准校验。

底层激活状态监测：为了识别更隐蔽的任务漂移，微软提出了比思维链审计更底层的审查维度，即通过捕捉模型内部的激活差异[23]来检测LLM是否在多轮交互中悄然偏离了原始指令。

针对智能体在长时推理场景中容易出现的逻辑断裂与深度幻觉问题，中兴通讯提出AI智能体框架Co–Sight[24]。该框架创新设计了冲突感知元验证（CAMV）与基于结构化事实的可信推理（TRSF）两大核心机制，将推理过程转化为可证伪、可审计的结构化流程，强制所有推理步骤基于来源验证、全链路可追溯的知识体系展开，从而从底层逻辑上规避了臆想结论与推理不一致性问题。该框架在通用AI助手（GAIA）基准测试中以87.04分夺冠，在人类终极考试（HLE）基准测试中以35.5分超越OpenAI、Google DeepMind同类框架，技术性能达到国际领先水平。

2）权限维度的流控制：防范泄露与越权

智能体中的数据泄露、越权操作等各类安全风险，最终都表现为信息流的违规，例如高敏感数据流向低权限主体。信息流控制（IFC）作为一种经典的安全模型被引入到智能体中，为处理多工具交互与实时决策等动态场景提供了一种可靠的访问控制策略。

动态标签与格模型：微软提出的FIDES框架[10]是将IFC机制与智能体架构深度融合的代表。该框架遵循“设计即安全”理念，对工具和数据标记敏感度与权限级别，强制执行“高敏感数据不流向低权限主体”以及“低可信数据不修改高敏感状态”的准则。通过实时监控安全标签的传播，FIDES能选择性地隐藏可能干扰规划器的数据，强化工具调用安全。类似地，谷歌的CaMeL框架[9]也在双模型架构与代码化执行的基础上集成了IFC，将工具与数据纳入安全格模型进行标签化管理，从架构层面阻断违规信息流。

为了在复杂的动态交互中实现更细粒度的策略管控，中兴通讯联合香港科技大学提出了智能体安全防御框架SEAgent。该框架构建了动态信息流图，实时追踪智能体、工具及数据库间的数据流转行为。结合强制访问控制（MAC）和基于属性的访问控制（ABAC）机制，SEAgent基于“首匹配原则”执行确定性规则，有效消除了概率性模型带来的不确定性风险，并通过灵活的策略配置满足不同业务场景下的差异化安全诉求。

2.5.3 执行层防护：工具行为监控与沙箱

执行层是智能体介入物理世界和数字系统的“手脚”，也是安全防御的最后一道防线。当恶意指令突破了感知层的过滤与决策层的审查后，执行层防护的核心目标转变为物理阻断与爆炸半径控制——即通过实时监控阻断危险动作，利用隔离环境限制破坏范围，确保即使智能体被劫持，也无法对宿主机或关键资产造成实质性损害。

1）逻辑侧：工具行为监控

工具行为监控聚焦于API调用序列的合法性检测，旨在识别并拦截“虽符合语法但违背业务逻辑”的异常操作。现有两类主流技术路线：

静态规则与正则匹配：基于传统静态正则规则的匹配方案，通过预设的黑名单规则库，精准拦截高危的Bash命令（如rm–rf）、特殊字符注入或涉及个人身份信息（PII）的违规传输。该方案部署成本低，但泛化能力弱，易被攻击者通过变种手段绕过。

动态上下文感知：为了应对更复杂的攻击，业界正转向基于上下文的动态检测。AgentArmor[25]提出将智能体的多步执行轨迹建模为程序依赖图，通过污点分析技术追踪不可信输入数据的传播路径，防范越权工具调用。Invariant[26]则聚焦智能体行为的上下文关联性，构建了工具使用序列与数据流的关联规则模型，能够精准识别并阻断不符合预期的异常行为序列。

针对智能体在执行层面临的框架异构性与检测时延挑战，中兴通讯提出了全链路自适应防护架构。该架构在接入层采用双模部署（代理/嵌入）适配异构框架，利用Hook技术实现深层数据采集；在检测层构建分级协同引擎，通过融合静态规则、轻量化模型与深度大模型，建立“漏斗式”防御机制。这种设计旨在从架构层面解决传统防御手段在兼容性与性能上的瓶颈，为高并发智能体应用提供高可用的安全底座。

2）物理侧：执行环境隔离

作为防御的底座，沙箱技术通过资源隔离构建安全的“防爆室”，将第三方工具与智能体核心模块物理隔开，防止恶意代码逃逸。

内核级隔离：适用于高隔离需求场景。以E2B为代表的MicroVM技术基于轻量级虚拟机监视器实现强隔离。Google提出的gVisor[27]则通过在用户空间构建“虚拟内核”，拦截并校验所有系统调用（Syscall），提供了兼顾容器轻量级与虚拟机高安全性的解决方案。该方案已成为Kubernetes生态中智能体沙箱的主流选择[28]。

应用级隔离：适用于轻量化场景。WASM[29]基于栈式虚拟机，将代码编译为平台无关字节码，运行于严格受限的线性内存中。微软将其应用于智能体工具调用沙箱[30]，通过动态加载WASM组件，提供了轻量级的浏览器级安全防护。

面对智能体在执行环境上对极速启动与强安全隔离的双重苛刻要求，中兴通讯构建了多层级融合隔离基础设施，创新性地融合了轻量级微虚拟机（MicroVM）、安全容器与WASM等多种技术路径，以适配不同敏感级与资源需求的任务；同时，引入快照预热机制，有效解决了隔离环境初始化的冷启动瓶颈；实现了毫秒级启动响应与资源的高效复用，为AI Agent从“能说”到“能做”的可靠落地提供了关键的基础设施保障。

2.6 智能体安全测评：攻防验证与评估基准

智能体安全测评是连接“研发态”与“运行态”的关键质量门禁，也是企业量化潜在风险、支撑安全决策的核心环节。在智能体全生命周期防护体系中，测评不仅是对防御机制有效性的“体检”，更是推动系统持续迭代的反馈源。

从内容合规到行为安全的测评范式转变：随着智能体技术的演进，安全测评范式已从传统的“内容导向型”向“行为导向型”转变。传统LLM测评体系以内容合规等静态指标为核心，而智能体的自主性决策特征、工具调用权限及多轮交互场景，决定了其测评体系须实现对动态决策逻辑、执行行为合规性及跨系统交互安全性的全维度覆盖。二者的具体差异如表2所示。

智能体威胁评估基准：为了应对上述挑战，学术界与产业界已针对智能体的不同风险维度，构建了多层次、差异化的测评基准，形成了针对性的“攻防演练场”。AgentHarm[31]以恶意智能体的危害行为为核心，构建专项评估基准，实现对智能体针对性安全威胁的精准测评。AgentSafetyBench[32]设计多场景评估框架，聚焦智能体全运行阶段的安全风险，具备广泛的场景适配能力。AgentDojo[33]是面向智能体核心威胁间接提示注入攻击的专项评估工具，依托操作系统与应用仿真环境，实现对智能体文本输出、API调用轨迹及环境状态变迁的全维度监测；其评价体系同步考量智能体“拒绝有害请求”与“执行良性指令”的双重能力，支持新工具集与新攻击提示词的灵活扩展，为安全评估的技术迭代提供支撑。

在工程实践层面，中兴通讯构建了“任务创建–执行–报告”的自动化全流程测评闭环，旨在解决企业级应用中测评效率低、覆盖面窄的痛点。该平台目前已覆盖内容安全与产品安全两大维度，涵盖31类网信办标准的内容安全场景，以及提示词注入、越狱攻击等产品安全场景，并将在2026

表2　大模型安全测评与智能体安全测评对比

对比维度	大模型安全测评	智能体安全测评
核心对象	模型本身（单一组件）	包含模型、工具、环境、记忆的完整系统
测评焦点	模型输出的内容安全与合规性	除模型风险外，更关注系统在动态环境中的行为安全性、决策可靠性以及整体流程的可控性
攻击面	相对静态、集中于输入/输出	高度动态，覆盖规划、工具、记忆、多轮交互全链路
测评环境	静态数据集、标准问答	动态、可交互的仿真环境（如网页、邮件系统）
典型风险	数据泄露、有害内容生成、幻觉	提示词注入、工具滥用、越权操作、多步诱导攻击

年升级为智能体评测平台，为智能体安全应用与监管提供高效、可靠的安全评估支撑。

3 中兴通讯智能体安全实践与探索

智能体安全风险不再是单点或静态的漏洞问题，而是一种贯穿输入、决策与行动全链路的动态威胁链。面对智能体带来的语义级注入、行为级越权等复杂攻击，传统的网络安全范式已难以为继。企业要建立有效的防护体系，必须从风险根因入手，识别智能体运行机制中的关键攻击面，构建覆盖语义安全、协议安全、决策与执行安全的立体化防御框架。

基于第二章提出的“全生命周期治理（SDLC）”与“纵深防御”理论体系，并结合深厚的行业实践经验，中兴通讯针对办公智能体、代码智能体、智能运维及具身智能等核心场景，提出了一套端到端的智能体安全防护方案。该方案将前文所述的架构级、模型级及运行时防护技术进行了工程化收敛与落地，构建了“大模型内生安全—智能体动态护栏—数据隐私底座”的3层纵深防护机制（如图4所示），旨在实现智能体的决策可信、行为可控、数据安全与风险可观测。

第1层防御：大模型安全防护

中兴通讯已构筑覆盖端云、支持多模态的大模型全方位安全防护体系，形成较为完备、业内领先的端到端安全防护屏障，累计已在智算一体机、家端智慧中屏、DT、iGPT等产品上线，商业化落地效果显著。

在模型训练阶段，通过敏感词库建设、监督微调（SFT）和基于人类反馈的强化学习（RLHF）安全对齐等技术，对自研模型进行内容安全训练，从源头构建模型的内生安全能力。

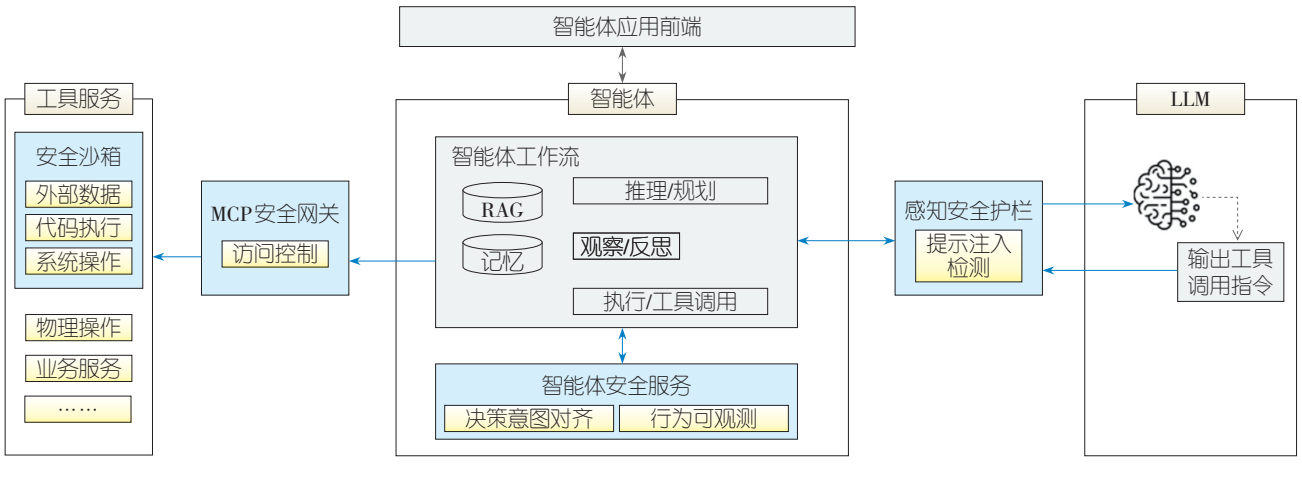
在推理运行阶段，大模型安全围栏在云端和端侧实现高性能、轻量化的安全检测，覆盖30多个风险类别，防护效果显著优于业界同类产品。云端采用参数量仅3B的模型，平均准确率优于参数量更大的Qwen3-VL-4B模型；同时，端侧轻量化版本的内存占用低于500 MB，在输入token长度为20的场景下平均时延仅5 ms，准确率达95%，检测性能较业界同类方案提升22%。

第2层防御：智能体安全护栏

智能体安全护栏是智能体安全防护方案的核心模块，其整体架构如图4所示，实现了对智能体运行全流程的深度管控。针对前文提到的执行层异构性与性能挑战，本方案在工程落地中取得了显著成效：

灵活部署与解耦设计：支持代理和嵌入两种模式，代理模式兼容OpenAI和模型上下文协议（MCP），实现护栏与智能体的解耦，适配不同技术框架。嵌入模式则深度适配通用的CrewAI框架及中兴通讯自研NAE平台Zagents_framework框架，通过Hook技术无感植入，实现了对原生智能体业务的零侵入保护。

协同防护引擎：采用“安全大模型 + 专用小模型 + 规则”的漏斗式混合架构，精准拦截提示词注入、恶意指令执行、隐私泄露等攻击，有效防护智能体工具调用、任务流安全威胁、决策意图操控、DDoS循环调用、敏感信息防护等多种典型安全威胁，兼顾性能与精度。实测数据表明：基于内存匹配的静态黑名单规则（如高危Bash指令拦截）检测时延低于1 ms，对业务运行几乎无感；轻量化检测小模型



LLM：大语言模型　MCP：模型上下文协议　RAG：检索增强生成

图4　中兴通讯智能体安全护栏

（针对恶意URL、SQL注入等）推理时延控制在10 ms级；针对复杂语义攻击的大模型深度审查，综合检测准确率达到95%以上。

全链路行为管控：通过Agent框架Hook技术采集全工作流数据，结合安全沙箱、MCP安全网关和感知安全护栏，实现对智能体工具执行、API访问、输入输出内容的细粒度审查和管控。

可观测与分析能力：配套智能体安全分析服务和可观测服务，实现实时审计、威胁处置与可视化监控。

第3层防御：数据隐私与合规保障

作为智能体安全防护的重要举措，中兴通讯通过数据隐私与合规保障体系的技术协同实现“合规性、隐私性、安全性”的三重目标。在人工智能生成内容（AIGC）内容合规层面，采用显式与隐式结合的数字水印技术，为生成式图片、音频及视频内容嵌入显式或隐式数字水印，严格契合网信办相关监管要求，通过轻量化设计实现性能影响最小化；在数据隐私处理层面，融合联邦学习、安全多方计算等前沿算法，构建包含40余种算子的脱敏工具集，覆盖主流脱敏场景；在底层安全支撑层面，引入AI可信执行环境（TEE）技术，依托硬件级隔离与加密机制，为智能体的敏感数据运算、模型推理流程及核心指令执行提供可信执行空间，从底层阻断非法访问与数据窃取，形成“软件脱敏+硬件隔离”的双重隐私防护，全方位保障数据使用过程中的合规性、机密性与完整性。

中兴通讯在智能体安全领域的关键技术路径上持续深耕，通过技术创新与工程化落地的深度融合，实现了从核心技术突破到高质量商业化应用的完整闭环。在大模型与智能体安全生态的构建进程中，上述工程实践正在验证“全生命周期纵深防御”理论的有效性。我们期望相关研究能为行业安全标准的完善与企业级智能体的规模化落地，提供有益的技术参照与实践样本。

4 结束语

智能体是AI规模化落地的核心载体，其安全防护至关重要。当前智能体安全领域已在内容安全、架构防御、行为防护等诸多方面取得重大进展，但仍存在诸多亟待解决的难题，如难以防护自动化的AI对抗型攻击、安全与系统灵活性矛盾突出、行业统一的安全评估标准缺失等。

面向未来，智能体安全防护需朝着技术创新、机制协同、生态共建的方向持续突破。在核心技术层面，应深化“以AI治理AI”的协同防御模式，提升对自适应对抗型攻击的主动识别与自主修复能力；同时强化可解释性技术与安全防护的融合，破解智能体决策黑箱难题，实现安全风险的精准溯源与责任界定。在场景适配层面，需构建“场景化规范–动态策略映射”框架，结合AI技术实现安全策略的自主迭代，缓解安全与灵活性的矛盾，适配办公、金融、医疗等关键领域的差异化需求。在体系构建层面，加快制定智能体安全技术标准与评估体系，建立威胁情报共享机制与开源安全生态，实现技术创新、标准规范与产业落地的良性循环。未来的智能体安全防护，不是建立一个绝对安全的“数字围墙”，而是构建一个具备免疫力、恢复力和适应力的安全生态。只有当安全体系能够主动识别、响应和修复风险时，智能体才能在金融、医疗、制造等关键产业中安全落地、稳健运行，实现真正的可控智能与可持续发展。

致谢

感谢香港科技大学王帅教授对本研究的帮助！同时感谢中兴通讯股份有限公司马苏安、武天元、吉鸿伟、金士英、蒋学鑫、殷玲玲、邓青伟等专家，在系统架构设计、全栈原型开发及综合评测过程中提供的宝贵建议与大力支持，他们的工程实践经验为本研究的理论落地提供了坚实基础。

参考文献

[1] Anthropic. Agentic misalignment: how LLMs could be insider threats [EB/OL]. (2025-06-21) [2026-01-05]. https://www.anthropic.com/research/agentic-misalignment
[2] OWASP. OWASP top 10 for agentic applications [EB/OL]. (2025-09-09) [2026-01-05]. https://genai. owasp. org/resource/owasp-top-10-for-agentic-applications-for-2026
[3] Del Rosario R F, Krawiecka K, De Witt C S. Architecting resilient LLM agents: a guide to secure plan-then-execute implementations [PP/OL]. arXiv(2025-09-10) [2026-01-05]. https://arxiv.org/abs/2509.08646
[4] Ji Z M, Wang X G, Li Z J, et al. Taxonomy, evaluation and exploitation of IPI-centric LLM agent defense frameworks [PP/OL]. arXiv(2025-11-19) [2026-01-05]. https://arxiv. org/abs/2511.15203
[5] Li E, Mallick T, Rose E, et al. ACE: a security architecture for LLM-integrated app systems [PP/OL]. arXiv(2025-04-29)[2026-01-05]. https://arxiv.org/abs/2504.20984
[6] Willison S. Prompt injection: what's the worst that could happen? [EB/OL]. (2023-04-14) [2026-01-05]. https://simonwillison. net/2023/Apr/14/worst-that-can-happen
[7] Kim J, Choi W, Lee B. Prompt flow integrity to prevent privilege escalation in LLM agents [PP/OL]. arXiv(2025-03-17)[2026-01-05]. https://arxiv.org/abs/2503.15547
[8] Wu F Z, Cecchetti E, Xiao C W. System-level defense against indirect prompt injection attacks: an information flow control perspective [PP/OL]. arXiv(2024-09-27) [2026-01-05]. https://arxiv.org/abs/2409.19091
[9] Debenedetti E, Shumailov I, Fan T Q, et al. Defeating prompt injections by design [PP/OL]. arXiv(2025-03-24) [2026-01-05]. https://arxiv.org/abs/2503.18813
[10] Costa M, Köpf B, Kolluri A, et al. Securing AI agents with

information-flow control [PP/OL]. arXiv(2025-05-29)[2026-01-05]. https://arxiv.org/abs/2505.23643
[11] Zhang J C, Yin L, Zhou Y, et al. AgentAlign: navigating safety alignment in the shift from informative to agentic large language models [PP/OL]. arXiv(2025-05-29)[2026-01-05]. https://arxiv.org/abs/2505.23020
[12] Wallace E, Xiao K, Leike R, et al. The instruction hierarchy: training LLMs to prioritize privileged instructions [PP/OL]. arXiv(2024-04-19)[2026-01-05]. https://arxiv.org/abs/2404.13208
[13] Chen S Z, Zharmagambetov A, Wagner D, et al. Meta SecAlign: a secure foundation LLM against prompt injection attacks [PP/OL]. arXiv(2025-07-03) [2026-01-05]. https://arxiv. org/abs/2507.02735
[14] OpenAI. Deliberative alignment: reasoning enables safer language models [EB/OL]. (2024-12-20)[2026-01-05]. https://openai.com/index/deliberative-alignment/
[15] Zhang J W, Estornell A, Baek D D, et al. Any-depth alignment: unlocking innate safety alignment of LLMs to any-depth [PP/OL]. arXiv(2024-10-20) [2026-01-05]. https://arxiv. org/abs/2510.18081
[16] Wang J X, Wu F Z, Li W D, et al. FATH: authentication-based test-time defense against indirect prompt injection attacks [PP/OL]. arXiv(2024-10-28) [2026-01-05]. https://arxiv. org/abs/2410.21492
[17] Wang Z L, Nagaraja N, Zhang L, et al. To protect the LLM agent against the prompt injection attack with polymorphic prompt [PP/OL]. arXiv(2025-06-06) [2026-01-05]. https://arxiv. org/abs/2506.05739
[18] Yi J W, Xie Y Q, Zhu B, et al. Benchmarking and defending against indirect prompt injection attacks on large language models [C]//Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1. ACM, 2025: 1809-1820. DOI: 10.1145/3690624.3709179
[19] Chen S Z, Wang Y Z, Carlini N, et al. Defending against prompt injection with a few defensive tokens [PP/OL]. arXiv(2025-07-10)[2026-01-05]. https://arxiv.org/abs/2507.07974
[20] Meta. Llama prompt guard 2 model card [EB/OL]. [2026-01-05]. https://github. com/meta-llama/PurpleLlama/blob/main/Llama-Prompt-Guard-2/86M/MODEL_CARD.md
[21] Microsoft. What is azure AI content safety? [EB/OL]. (2025-09-16) [2026-01-05]. https://learn. microsoft. com/en-us/azure/ai-services/content-safety/overview
[22] Inan H, Upasani K, Chi J F, et al. Llama guard: LLM-based input-output safeguard for human-AI conversations [PP/OL]. arXiv(2023-12-07) [2026-01-05]. https://arxiv. org/abs/2312.06674
[23] Abdelnabi S, Fay A, Cherubin G, et al. Get my drift? catching LLM task drift with activation deltas [PP/OL]. arXiv(2024-06-02)[2026-01-05]. https://arxiv.org/abs/2406.00799
[24] Zhang H W, Lu J, Jiang S Q, et al. Co-sight: enhancing LLM-based agents via conflict-aware meta-verification and trustworthy reasoning with structured facts [PP/OL]. arXiv(2025-10-24)[2026-01-05]. https://arxiv.org/abs/2510.21557
[25] Wang P R, Liu Y, Lu Y F, et al. AgentArmor: enforcing program analysis on agent runtime trace to defend against prompt injection [PP/OL]. arXiv(2025-08-02)[2026-01-05]. https://arxiv.org/abs/2508.01249
[26] Invariant Labs. Invariant [EB/OL]. [2026-01-05]. https://invariantlabs.ai/
[27] Google. gVisor [EB/OL]. [2026-01-05]. https://gvisor.dev
[28] Google Cloud. Isolate AI code execution with Agent Sandbox [EB/OL]. [2026-01-05]. https://docs. cloud. google. com/kubernetes-engine/docs/how-to/agent-sandbox
[29] WASI. The webassembly system interface [EB/OL]. [2026-01-05]. https://wasi.dev/
[30] Microsoft. Introducing Wassette: WebAssembly-based tools for AI agents [EB/OL]. (2025-08-06) [2026-01-05]. https://opensource. microsoft. com/blog/2025/08/06/introducing-wassette-webassembly-based-tools-for-ai-agents
[31] Andriushchenko M, Souly A, Dziemian M, et al. AgentHarm: a benchmark for measuring harmfulness of LLM agents [PP/OL]. arXiv(2024-10-11) [2026-01-05]. https://arxiv. org/abs/2410.09024
[32] Zhang Z X, Cui S Y, Lu Y D, et al. Agent-SafetyBench: evaluating the safety of LLM agents [PP/OL]. arXiv(2024-06-19)[2026-01-05]. https://arxiv.org/abs/2412.14470
[33] Debenedetti E, Zhang J, Balunović M, et al. AgentDojo: a dynamic environment to evaluate prompt injection attacks and defenses for LLM agents [PP/OL]. arXiv(2024-10-11)[2026-01-05]. https://arxiv.org/abs/2406.13352

作 者 简 介

闫新成，中兴通讯股份有限公司首席安全架构师，江苏省产业教授，正高级工程师；主要研究方向为5G/6G安全、AI安全；从事电信行业20年，曾主持国家科技重大专项课题，获得多项省部级科技奖励；拥有专利40余项。

刘东，中兴通讯股份有限公司副总裁、中心研究院副院长；主要从事操作系统和网络安全领域的技术研究和经营管理工作。

李旻旻，中兴通讯股份有限公司技术预研工程师；主要研究方向为主机入侵检测、AI内容安全、智能体安全、可信与机密计算等。

吴建华，中兴通讯股份有限公司技术预研工程师；主要研究方向为主机安全、智能体安全、AIGC反欺诈等。