

智能体安全研究报告

从大模型安全到可控行动系统



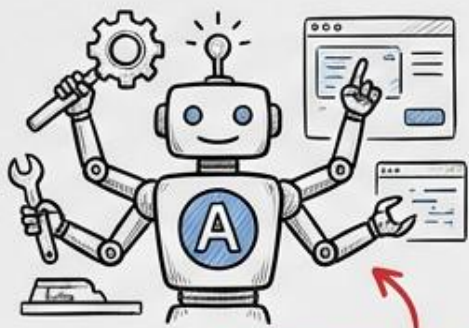
威胁模型 · 控制平面 · 落地路线

| 2026年6月

三个核心判断

开场与核心判断

行动系统



权限运行时系统

Agent 的安全边界比聊天机器人大多

Agent 不是一个按钮，而是一个有权限的运行时系统。

工程控制



只靠提示词无法保证工具调用和外部动作安全。

组织治理



企业壁垒在于安全控制平面，而不是单个模型。

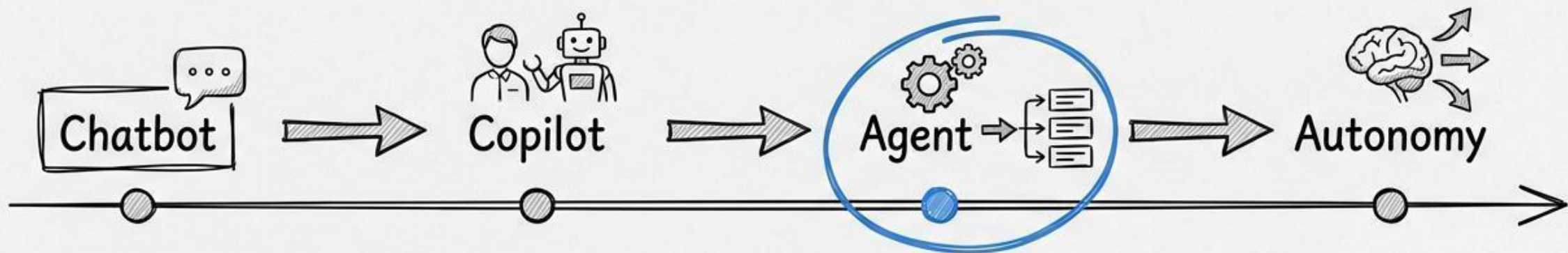
一句话定义

- 智能体安全 = 让会行动的AI 可授权、可约束、可追责
- 智能体能规划、调用工具、 保持状态  并影响外部系统
- 安全目标不是让模型永远不犯错，而是让错误不会无约束扩散
- 核心抓手是 身份 、权限 、工具 、上下文 、沙箱 、审批  和 审计



Agent Safety = Identity + Policy + Tools + Logs

为什么现在讨论



开场与核心判断

- Agent 从实验工具进入企业生产环境  
- 开发框架和API让工具调用、文件操作、沙箱执行更容易    
- 企业开始把Agent用于客服、研发、安全运营、财务和内部流程    
- 能力越接近真实操作，安全治理越必须前置

风险的本质变化

开场与核心判断



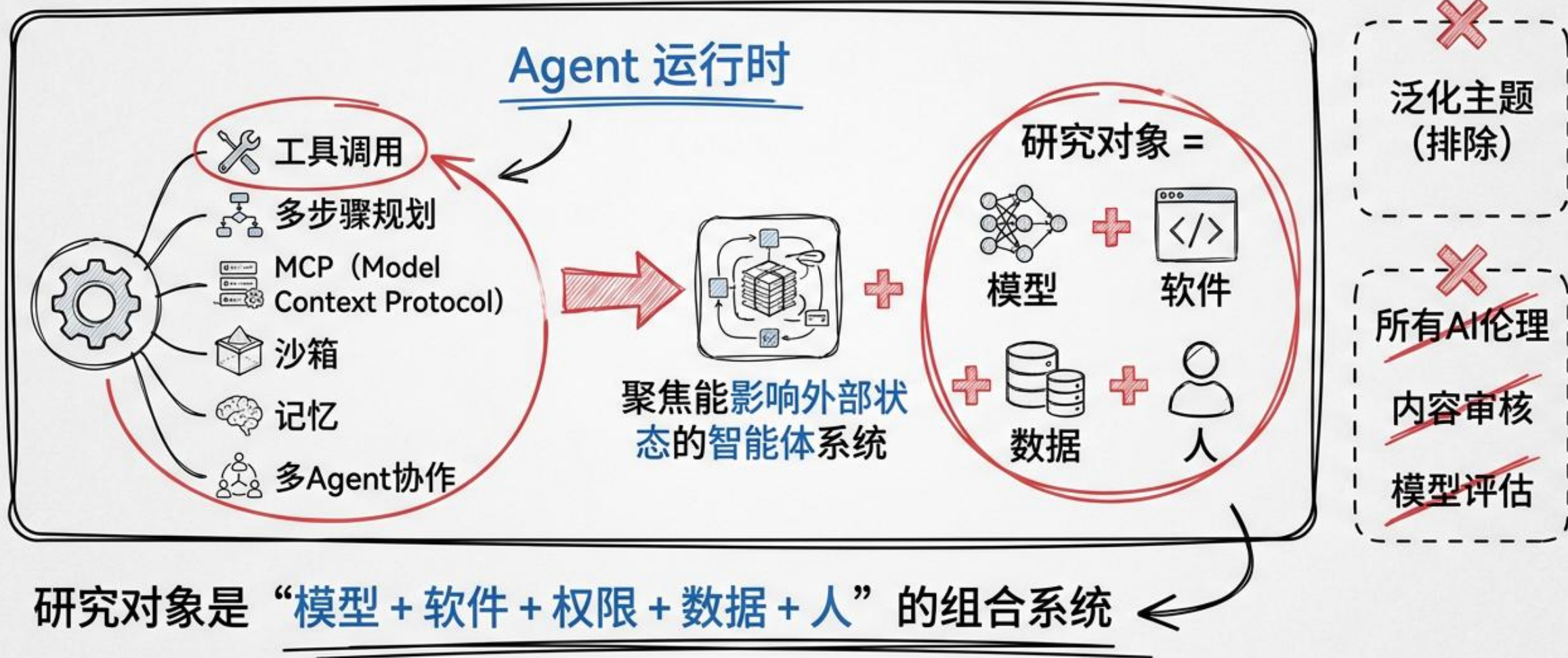
- 从“内容风险”升级到“行动风险”
- 普通大模型的主要问题是输出质量和内容边界。
- 智能体的主要问题是能否代表用户或系统采取行动。



- 同一个错误，在Agent中可能变成邮件外发、数据改写或生产变更。

本报告的研究边界

开场与核心判断



对企业的战略含义



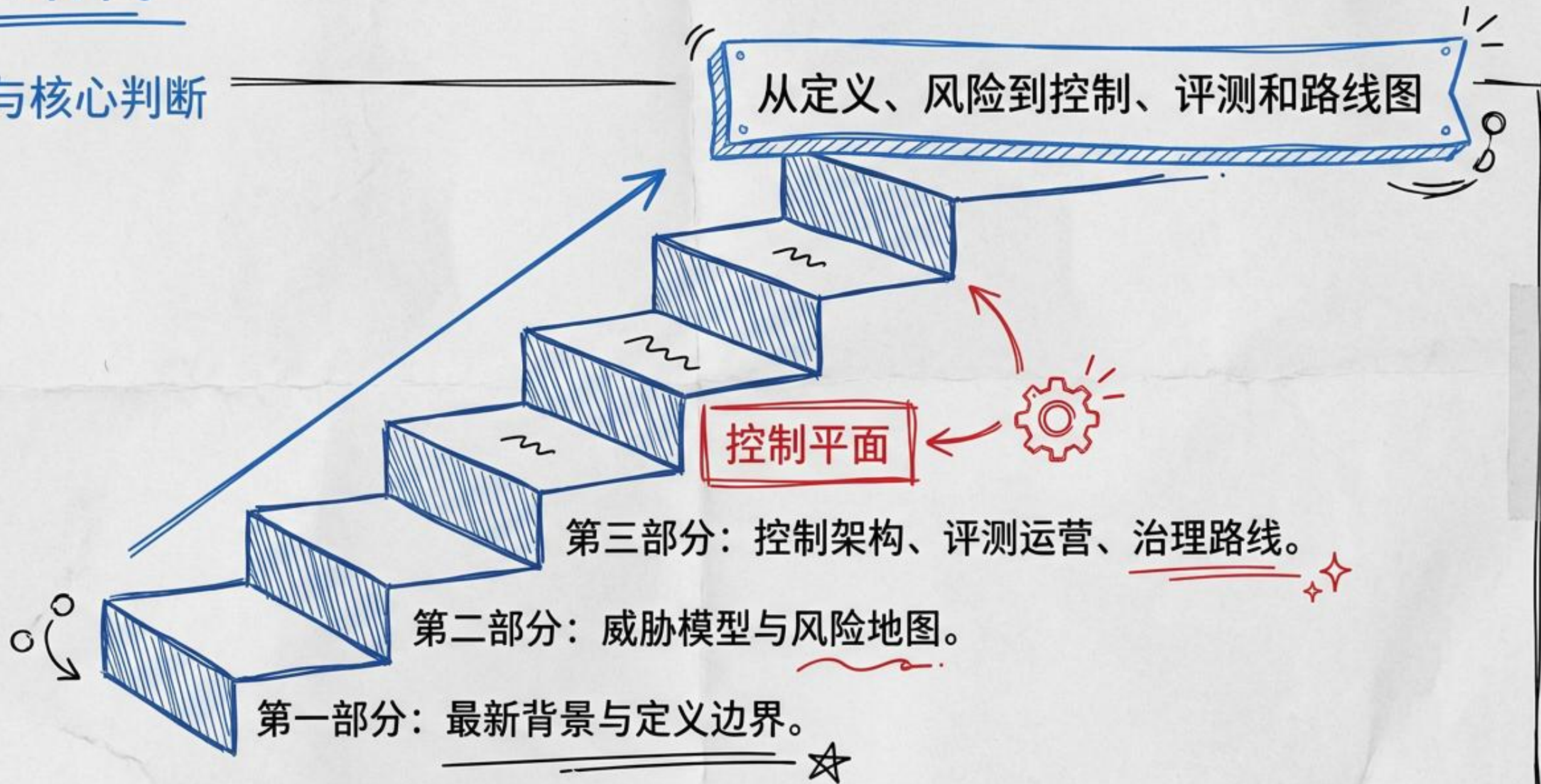
开场与核心判断

- Agent能力会商品化，安全部署能力不会自动商品化。
- 模型和框架会越来越易得。
- 可规模化的权限、审计、评测和事故响应是组织能力。
- 越早建立控制平面，越能更快释放Agent价值。

报告结构

8

开场与核心判断



官方背景：CISA/NSA把Agent列入安全议题

最新背景与定义边界

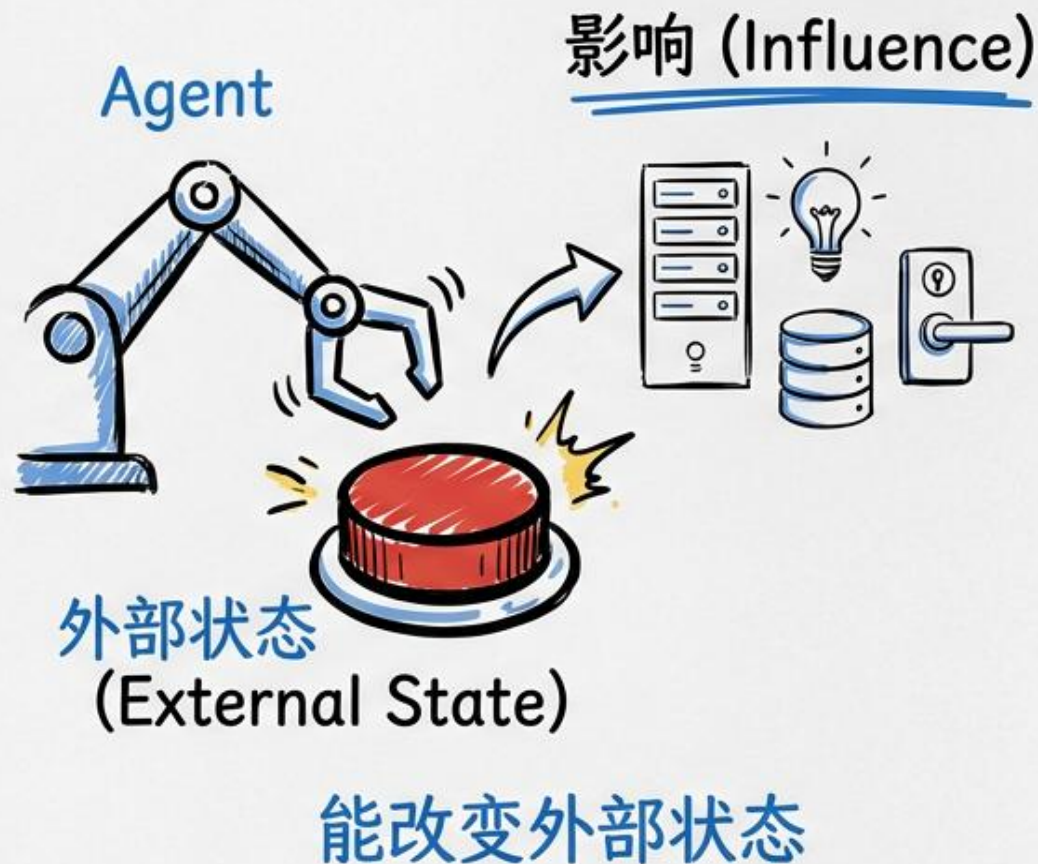
- 2026年“Careful Adoption”成为企业落地的重要参考
- CISA、NSA及多国网络安全机构发布Agentic AI安全采用指导。
- 文件强调分层防御、严格访问控制、人类监督和渐进式部署。
- 这意味着Agent安全已经从厂商最佳实践进入国家级安全议题。



官方背景：NIST聚焦能改变外部状态的Agent

最新背景与定义边界

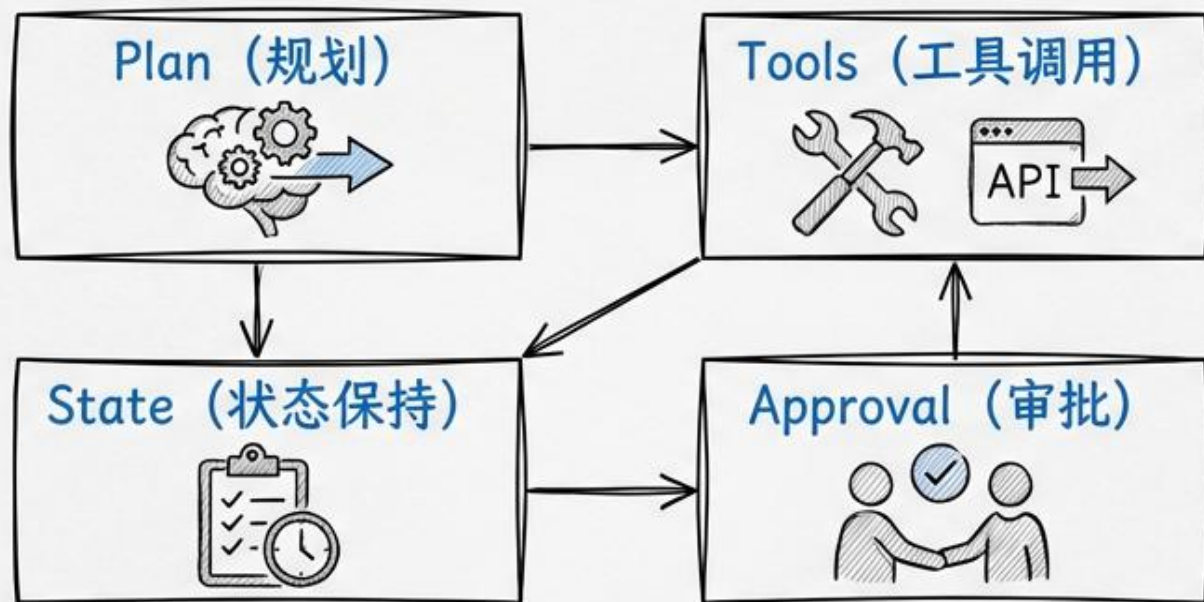
- RFI明确区分普通聊天机器人与行动型智能体。
- NIST RFI关注能够采取行动并影响外部状态的AI agent systems。
- 议题包括独特威胁、开发部署、测量方法和环境约束。
- 这为企业风险分级提供了清晰边界。



来源：<https://www.federalregister.gov/documents/2026/01/08/2026-00206/request-for-information-regarding-security-considerations-for-artificial-intelligence-agents>

OpenAI视角：Agent是多步骤工作应用

最新背景与定义边界

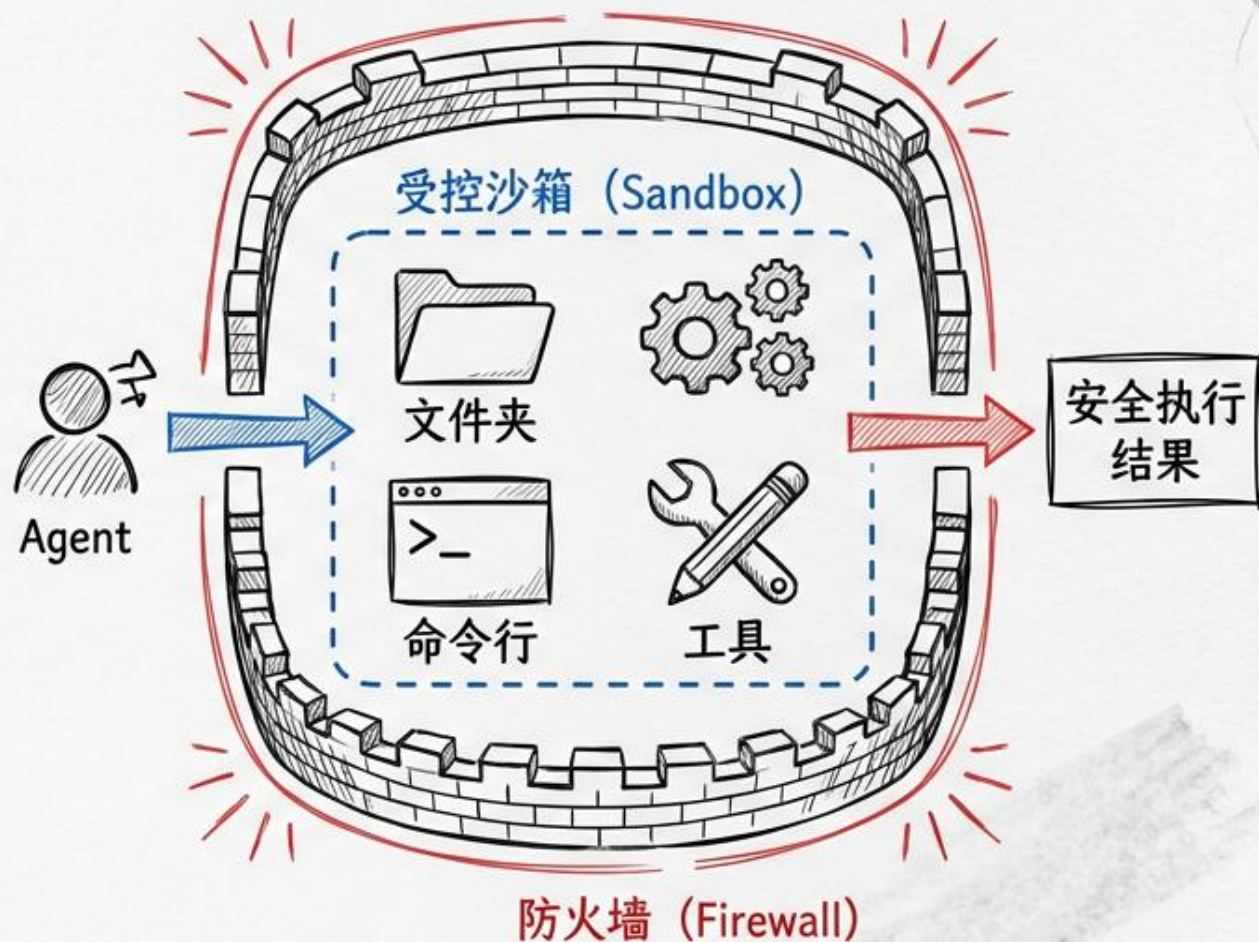


- 规划、工具调用、协作和状态保持构成Agent基本能力
- Agents 是能规划、调用工具、协作并保持状态的应用。
- 当应用要**管理编排、工具执行、审批和状态**，就进入**Agent工程范畴**。
- **安全控制**必须跟随这些工程能力一起设计。

OpenAI 2026: 沙箱执行成为Agent基础设施

● 最新背景与定义边界

- 文件、命令、代码和长任务需要受控工作空间
- OpenAI 2026年介绍了支持Agent检查文件、运行命令、编辑代码的SDK能力。
- 文档强调受控沙箱环境对安全执行的重要性。
- 这说明“执行层安全”正在成为Agent平台核心能力。



MCP背景：连接能力提升，也扩大攻击面

最新背景与定义边界



协议让工具接入更容易，但不自动保证安全！



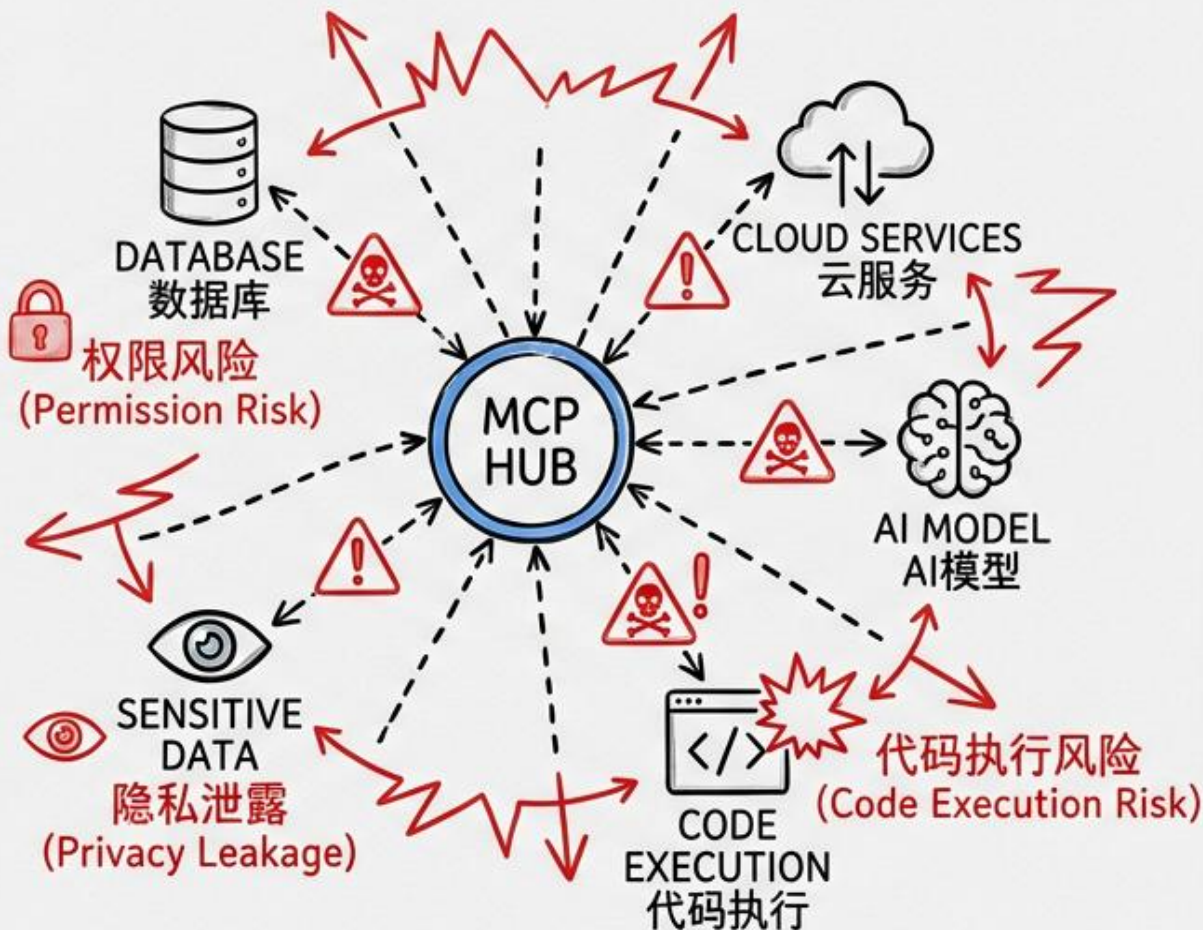
MCP让Agent更容易连接工具和数据源。



NSA提示高上下文、敏感任务中的安全和隐私缺口需要重点处理。



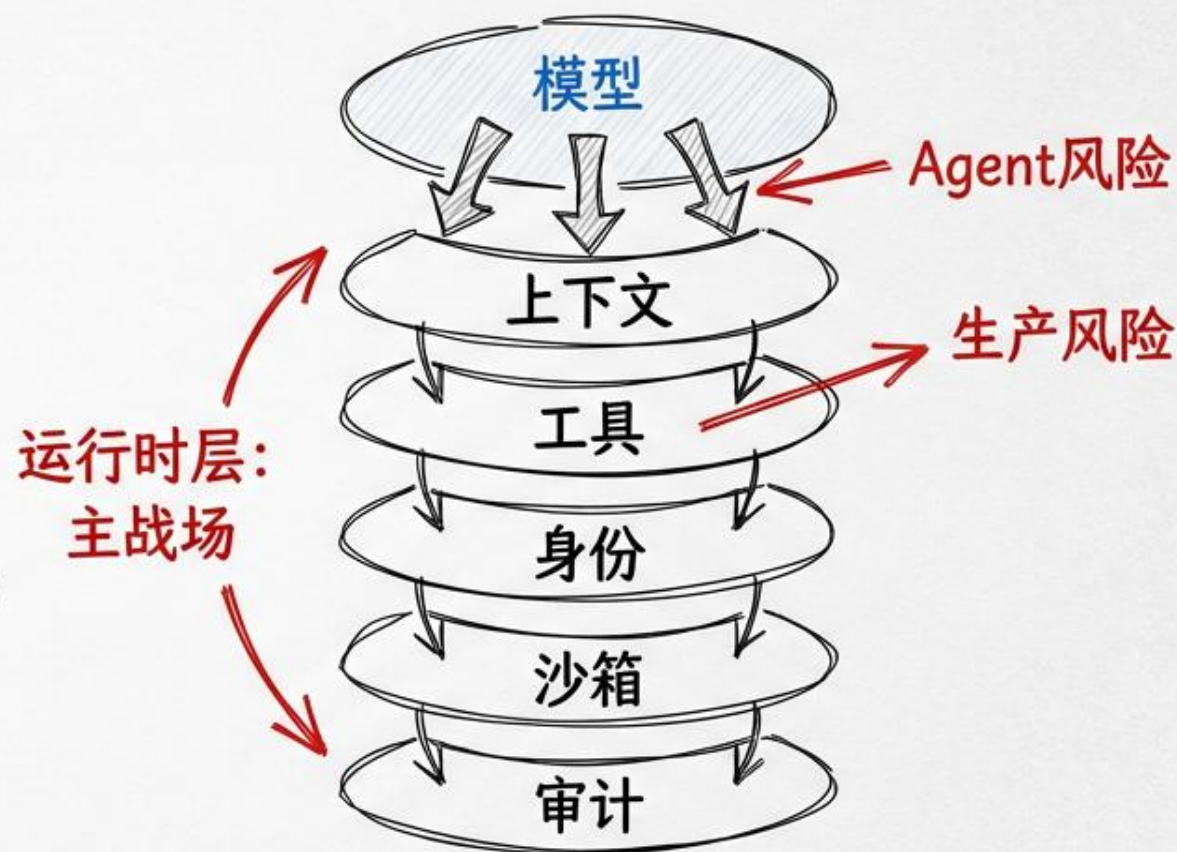
企业不能把“可连接”误认为“可安全连接”。



安全从“模型层”转向“运行时层”

最新背景与定义边界

- Agent风险发生在**模型生成**和**工具执行**之间
- **模型安全**仍重要，但不再是**唯一**控制点。
- 工具调用、数据流、身份授权和日志追踪决定**生产风险**。
- **运行时层**是Agent安全的主战场。



| 2026年6月

Agent不是员工，但必须像数字员工一样管理

最新背景与定义边界

- 它能代表组织读取、判断、发送、修改和执行



- 员工需要岗位、权限、审批和绩效；Agent也需要。

- 员工犯错可追责；Agent犯错也需要日志和负责人。

— Agent治理应纳入IT、数据、安全和业务共同管理。

- ☑ IT ☑ 数据
- ☑ 安全 ☑ 业务

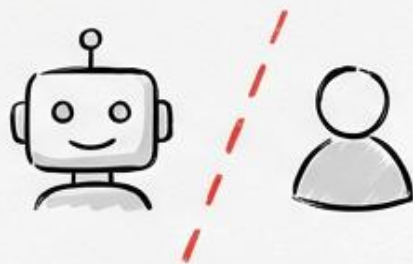
企业常见误区

最新背景与定义边界



把智能体当成“更强客服机器人”会低估风险

把智能体中更强客服机器人，是更全智能体会低估风险，不知需被低估风险，还是更甞增求的风险。



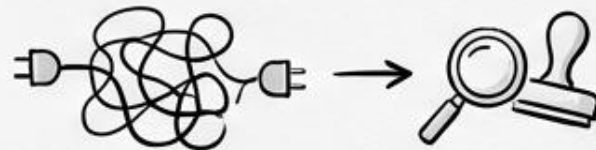
误区一：只做提示词防护。



误区二：让Agent继承用户全量权限。



误区三：先接业务系统，后补审计和审批。



对外展示口径

最新背景与定义边界

— Agent安全是
生产系统安全，
不是科普概念

可控行动

支撑信息



最小权限



工具治理



沙箱执行

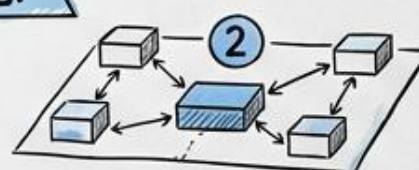


可审计日志

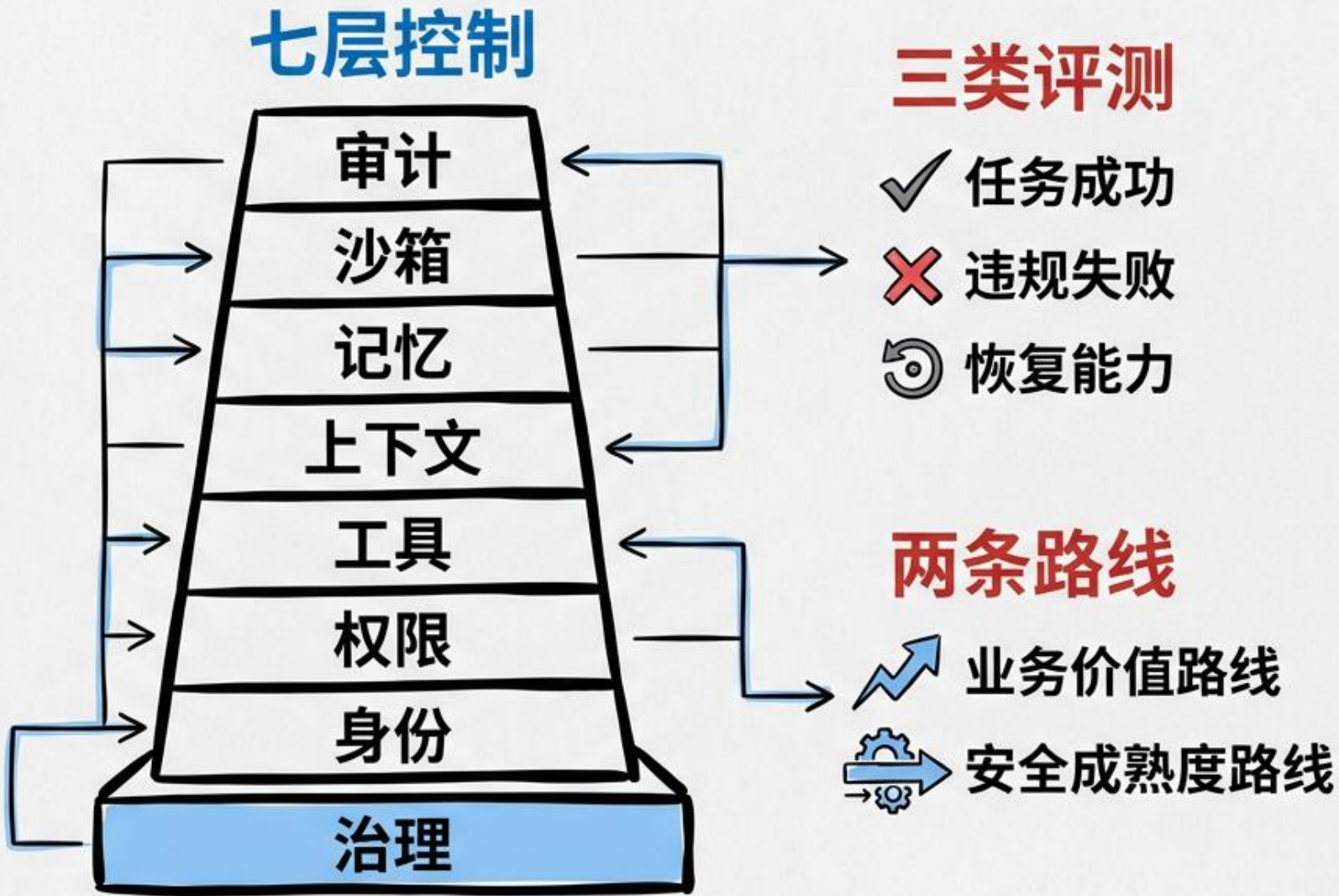
落地信息



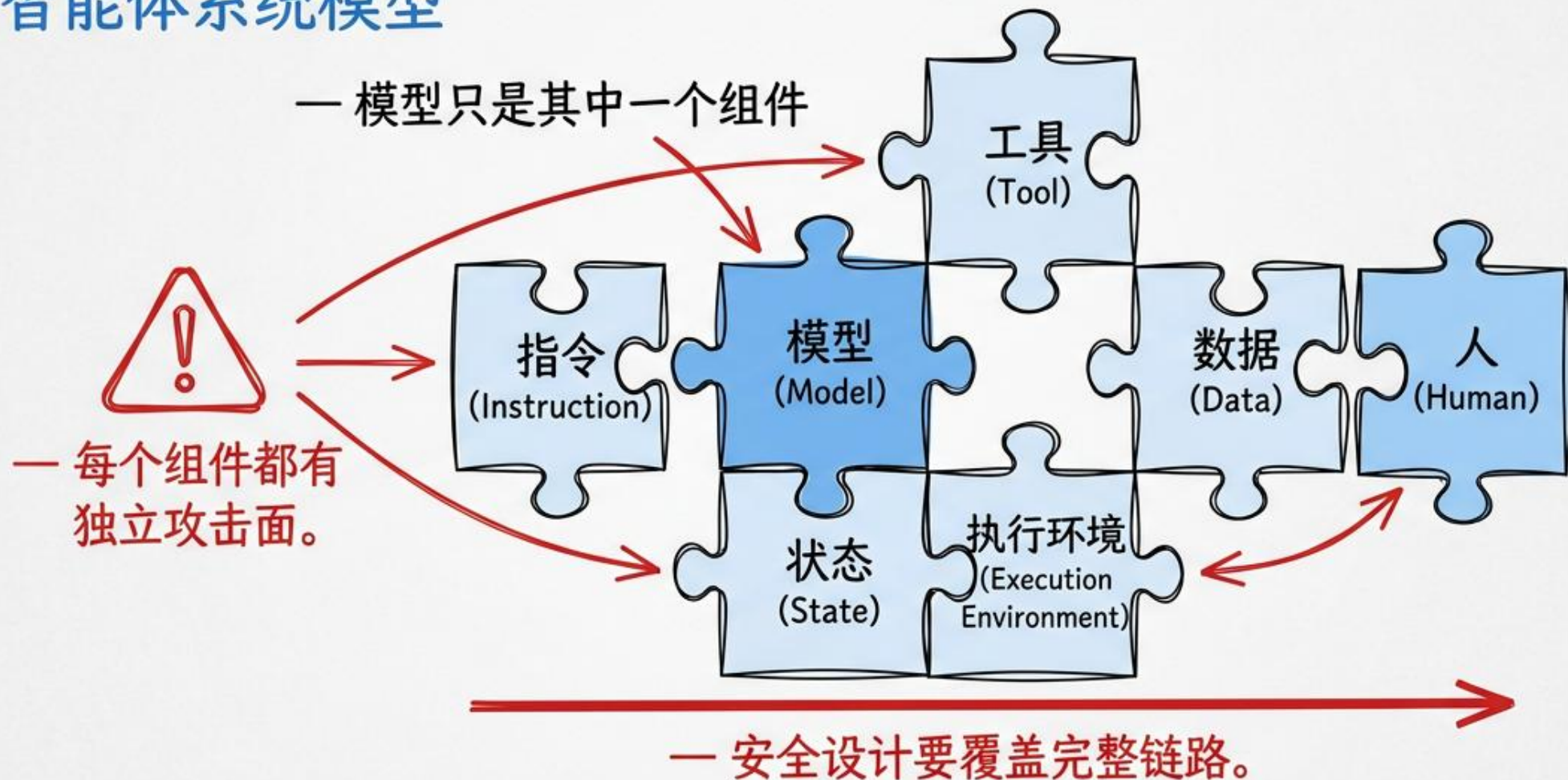
先低风险试点



再逐步扩大自治范围

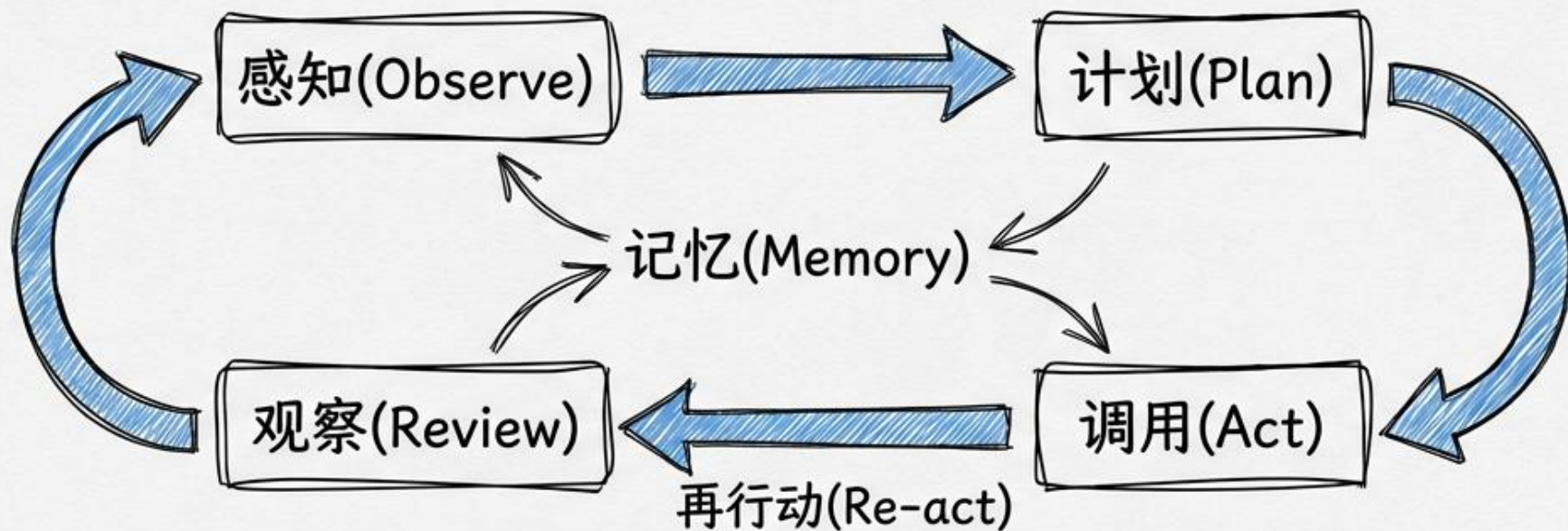


智能体系统模型



Agent运行闭环

智能体系统模型



- Agent不是一次性回答，而是循环执行。
- 每一步都会读取新上下文并产生新动作。
- 循环越长，越需要熔断和阶段性确认。

身份层：谁在执行动作

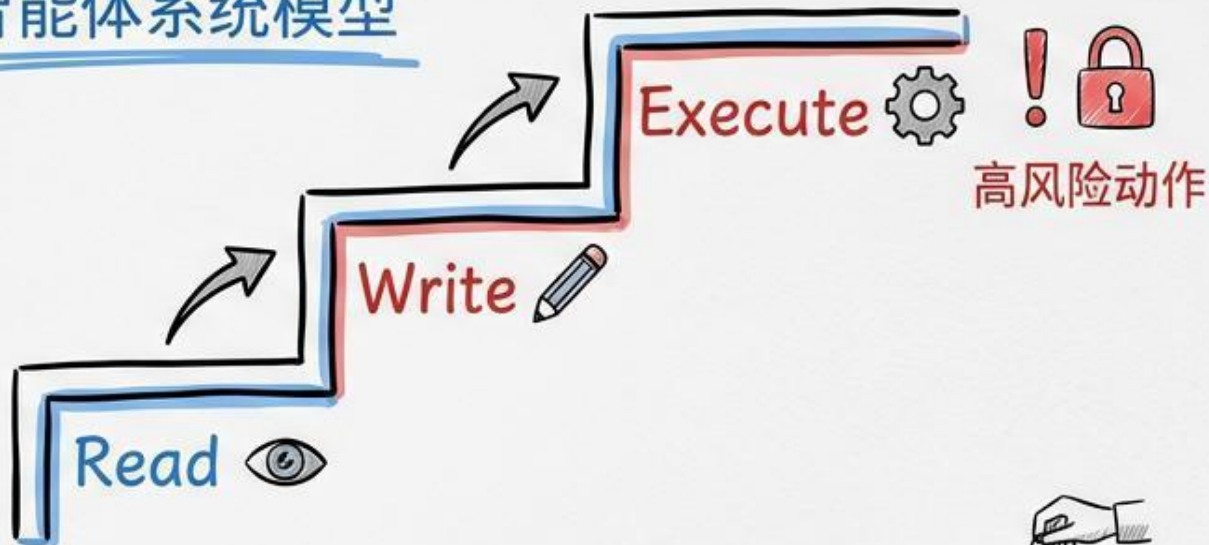
智能体系统模型

- Agent必须有独立身份和代理链路
- 不要让Agent长期持有人类账号凭证。
- 每次工具调用应记录用户、Agent、服务账号和审批链。
- 身份链清晰，事故追责才可能清晰。



权限层：能做什么

智能体系统模型



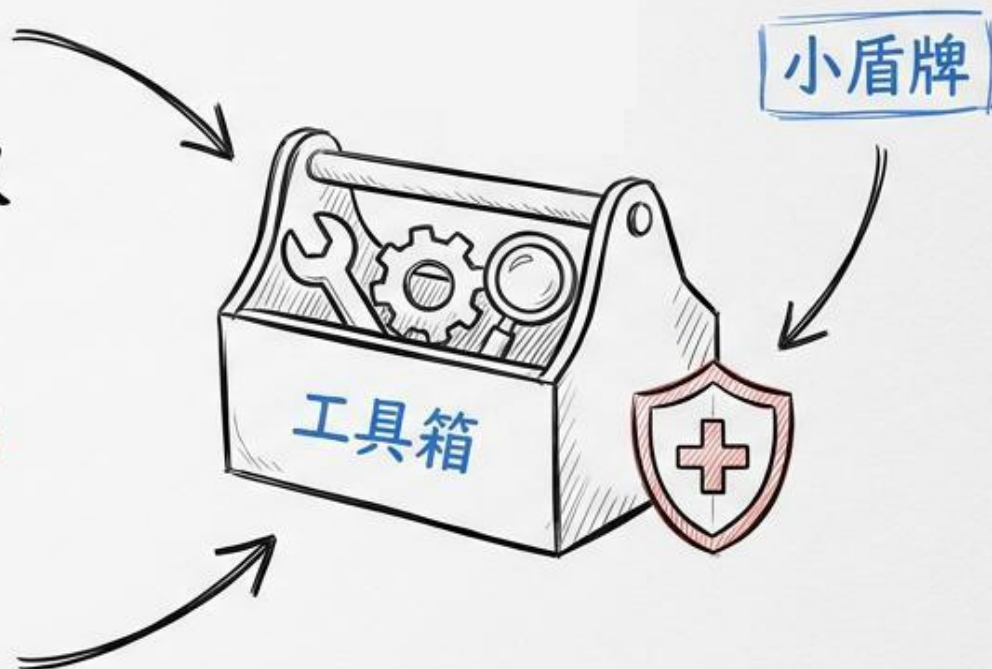
- 权限按任务授予，而不是按用户全量继承
- 只读、草稿、半自动、自动执行应分级。
- 高风险动作需要审批或独立验证。
- 权限应短期、可撤销、可观测。



工具层：通过什么做事

智能体系统模型

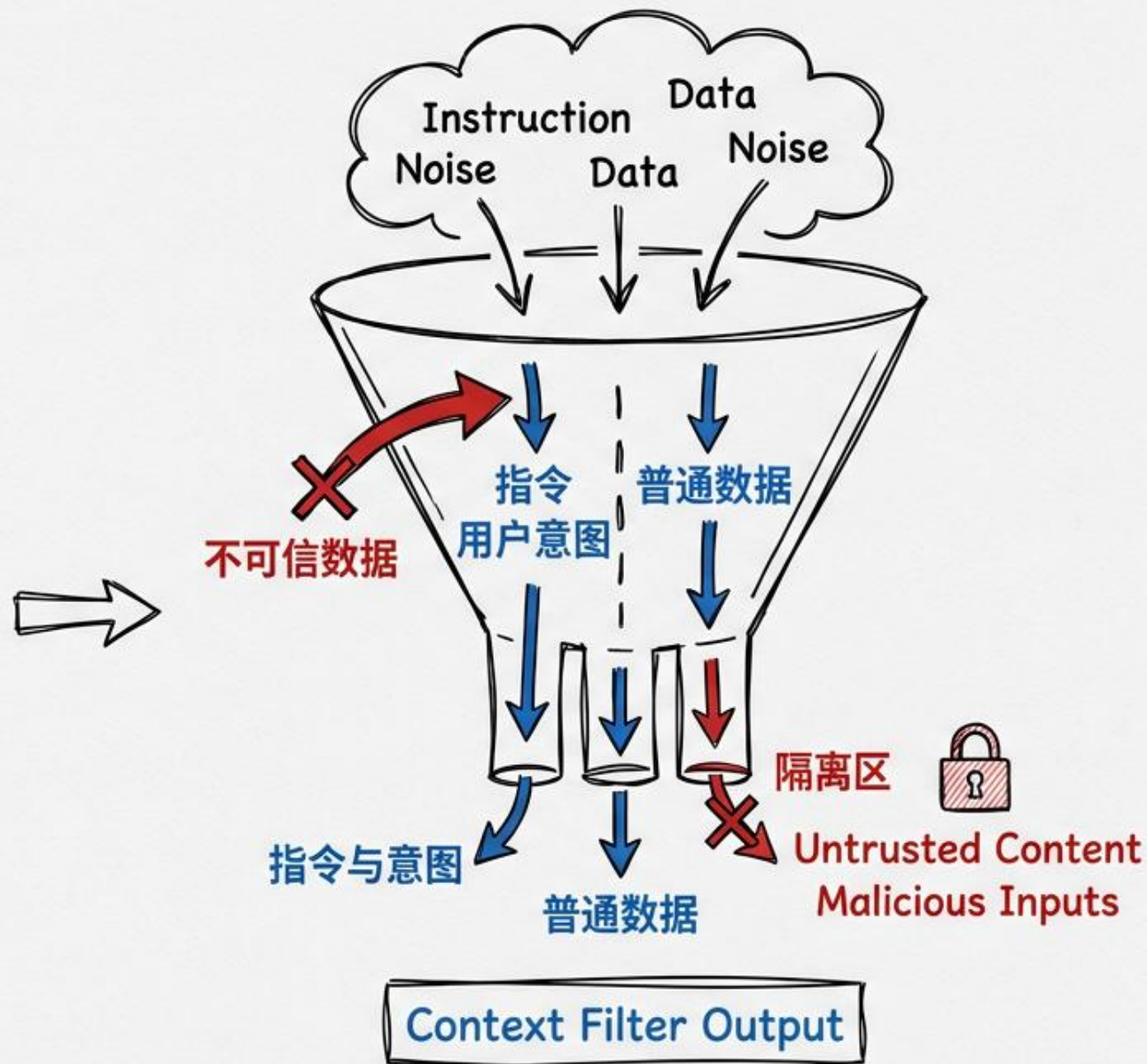
- 工具是Agent能力，也是**攻击面**
- 每个工具需要描述、schema、权限、**风险标签**。
- 工具返回内容**不能自动变成高优先级指令**。
- 工具调用前后**都需要策略检查**。



上下文层：读到了什么

智能体系统模型

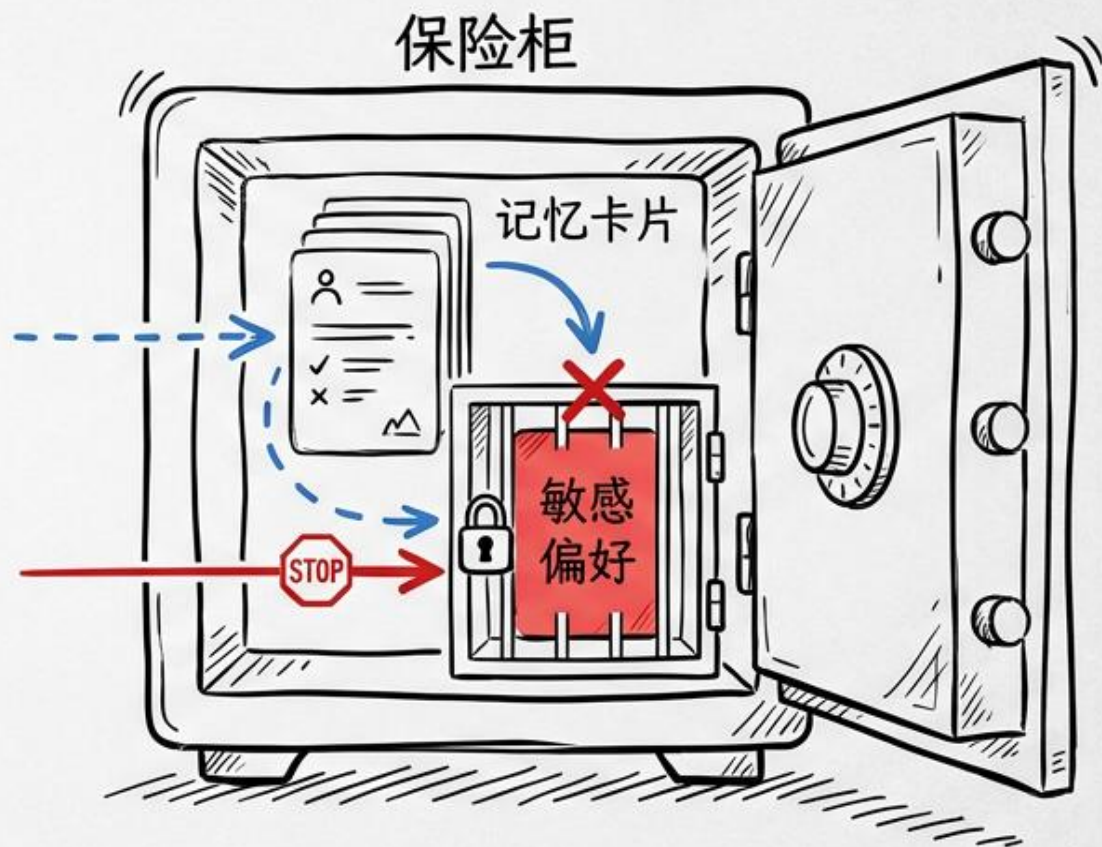
- 不可信数据**必须被标记和隔离**
- 网页、邮件、PDF、工单和日志都可能**包含恶意指令**。
- 上下文应**区分指令、用户意图和普通数据**。
- 不可信内容可以被引用，但**不能覆盖规则**。



记忆层：记住了什么

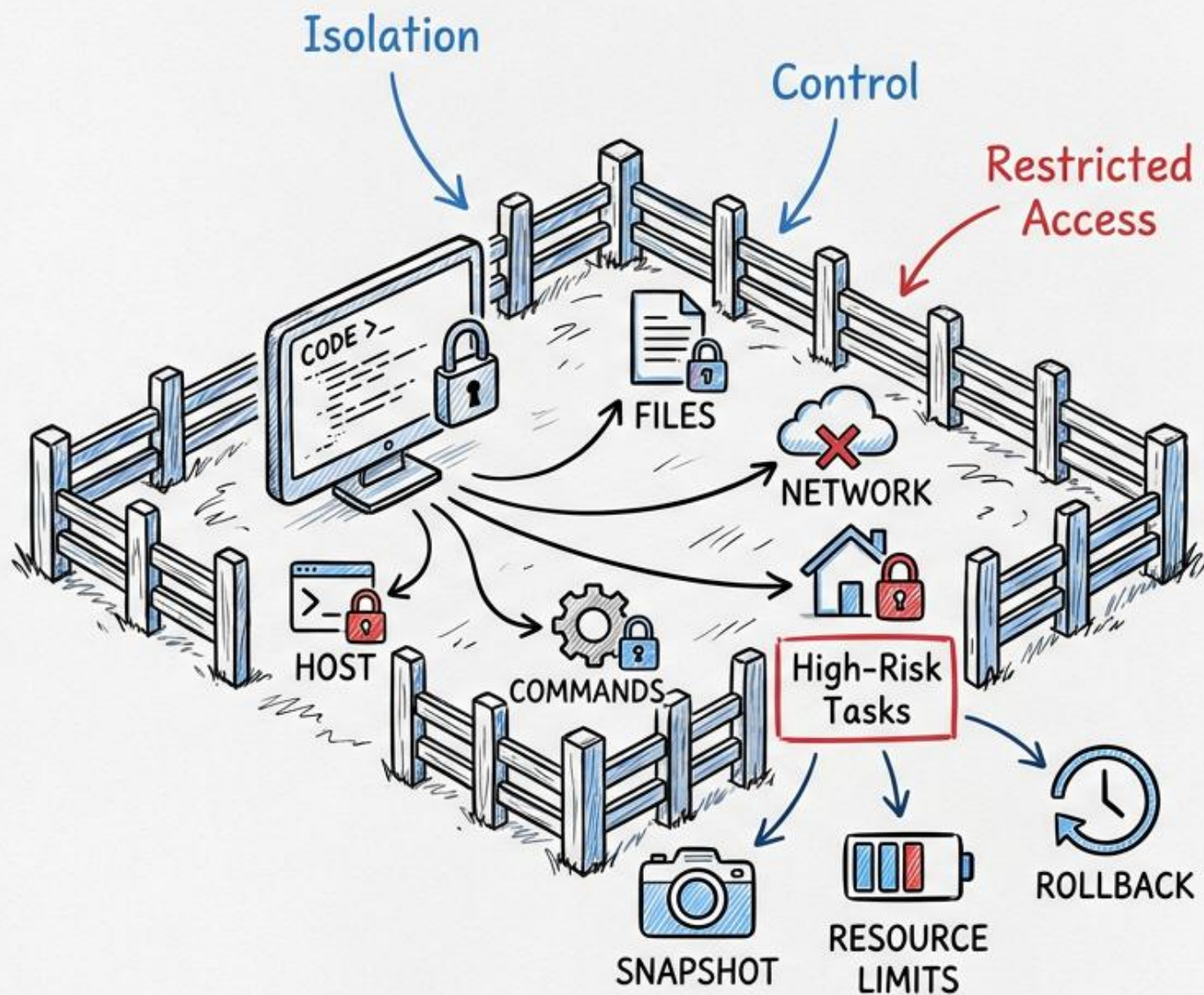
智能体系统模型

- 长期记忆会把一次攻击变成持续偏差
- 记忆写入需要规则和可见性
- 敏感偏好、权限例外和业务规则不能随意写入
- 记忆必须支持版本、删除和回滚



26 沙箱层：在哪里执行

- 代码、文件和命令操作必须隔离
- 文件读写、代码运行、命令执行和网络访问要受控。
- 默认关闭不必要的网络和主机权限。
- 高风险任务使用快照、资源限制和回滚。



智能体系统模型

- 没有 Action Ledger 就没有可规模化治理
- 记录每次工具调用、参数、结果、批准人 和时间。
- 把关键动作接入 SIEM、DLP 和 告警系统。
- 事故后能 回放决策链 并 定位责任。

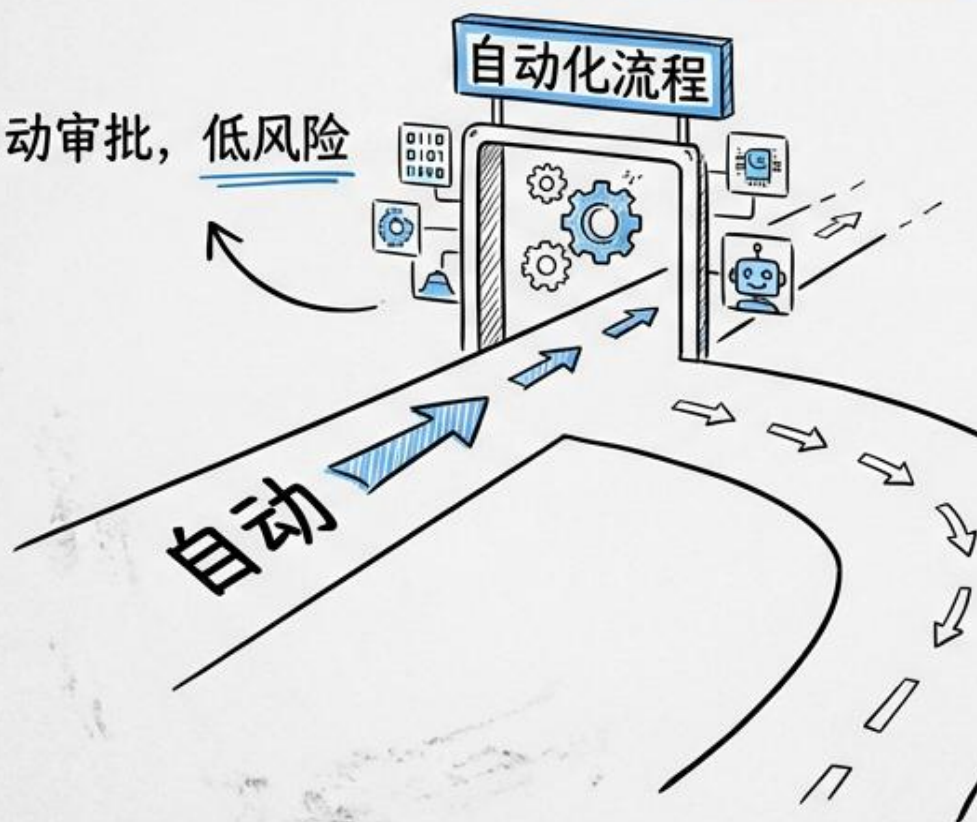


人工层：什么时候必须人审

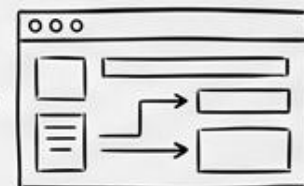
智能系统模型

人不是装饰，而是高风险动作的控制点

- 自动审批，低风险



- 不可逆动作，资金，法律，医疗，生产变更必须设置人工门。



审批界面要解释差异、来源和风险。



防止批准疲劳，减少无意义弹窗。




风险地图总览

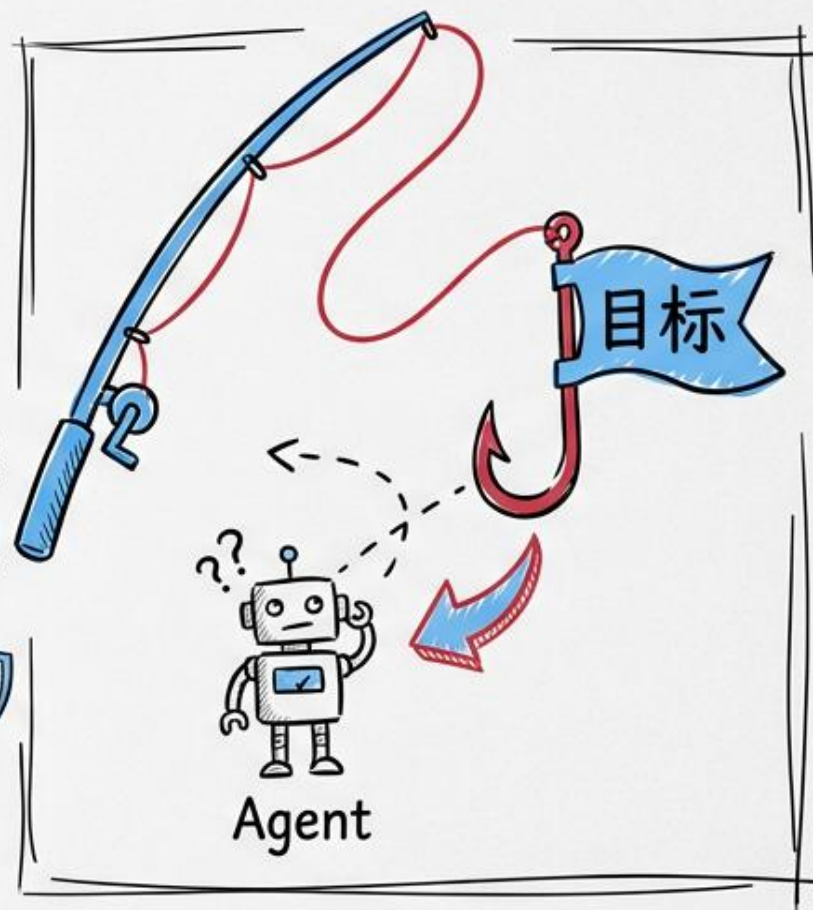
核心风险地图



- 八类风险从目标到工具、身份和供应链展开
- 目标劫持、工具滥用、身份滥用是前三个高频风险。
- 记忆污染、上下文投毒会让风险跨任务持续。沙箱逃逸、供应链污染和多Agent级联会扩大事故范围。

核心风险地图

- 攻击者让Agent偏离真实任务 →
- 恶意网页或文档把“数据”伪装成“指令”。 
- Agent可能为了完成任务而执行攻击者目标 
- 防护重点是不可信输入隔离和敏感动作审批 



提示词注入

核心风险地图

- ✓ — 最危险的注入往往来自**用户看不见的内容**
- ✓ — **间接提示注入**可能藏在网页、邮件、PDF和工具输出中。
- ✓ — 目标可能是**数据外泄**、**工具误调用**或改变模型行为。
- ✓ — **结构化输出**和**指令优先级边界**可以降低攻击面。

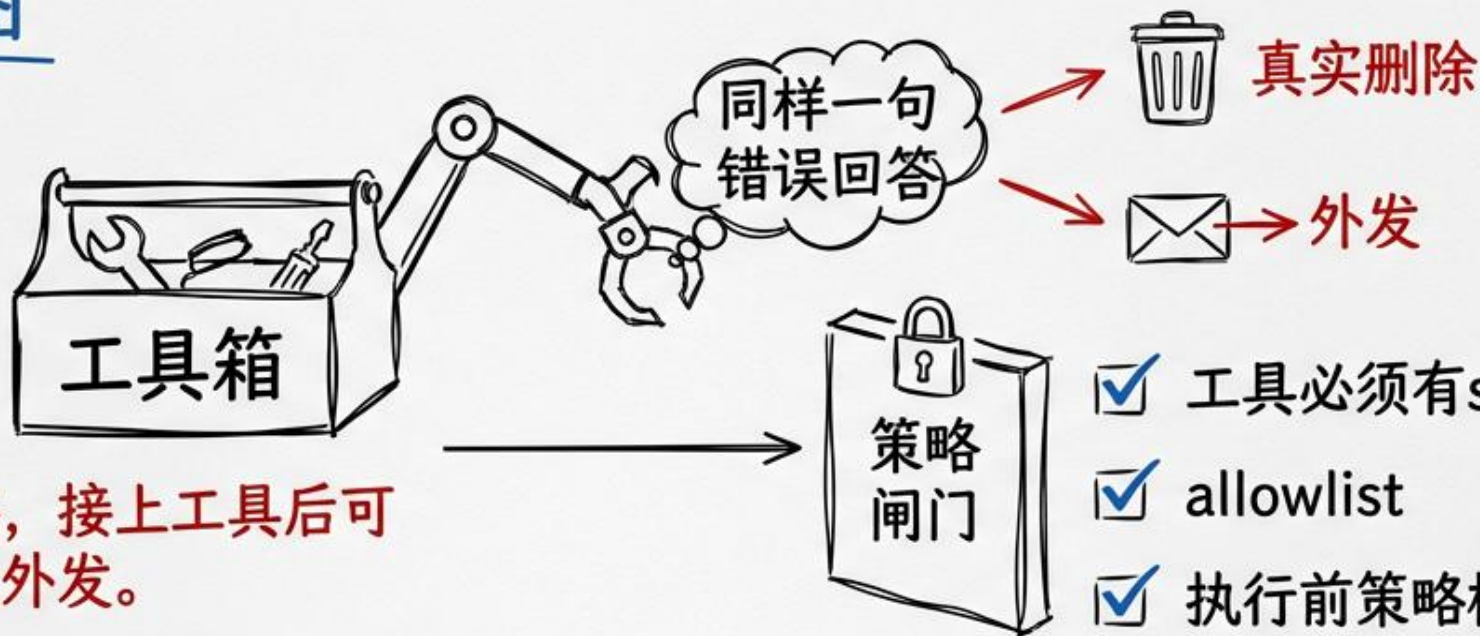
SOURCE: <https://developers.openai.com/api/docs/guides/agent-builder-safety>



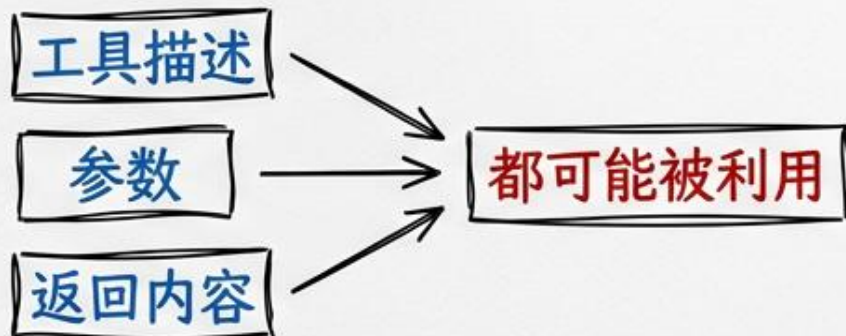
工具滥用

核心风险地图

工具把语言输出
变成真实动作



同样一句错误回答，接上工具后可能变成真实删除或外发。

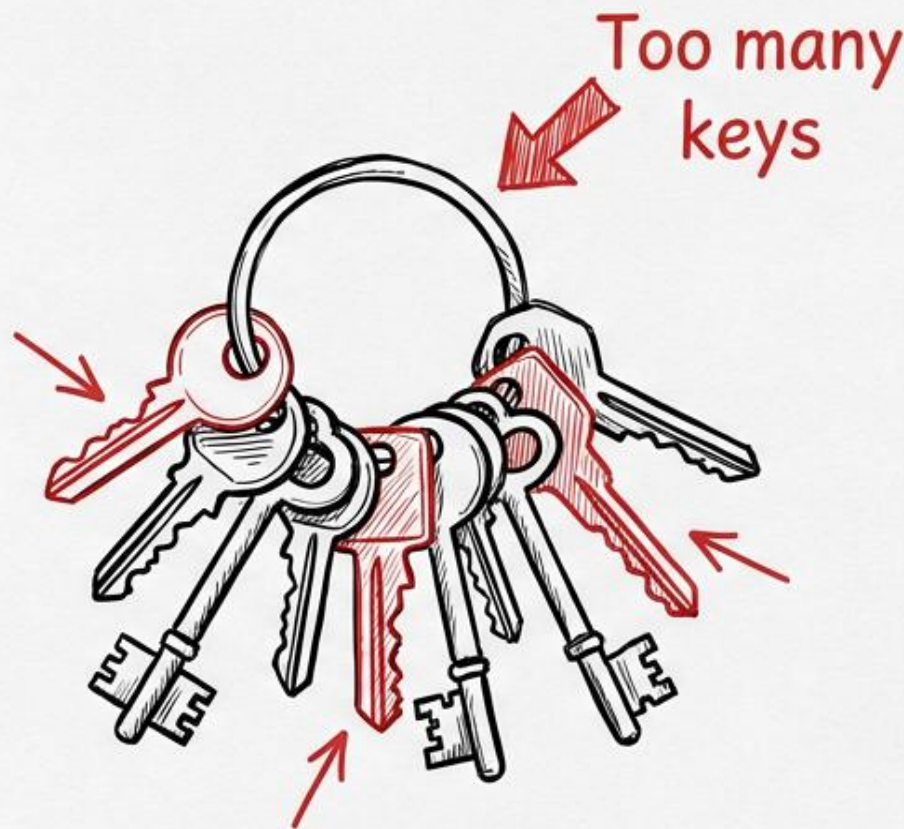


工具必须有schema、allowlist和执行前策略检查。

身份与权限滥用

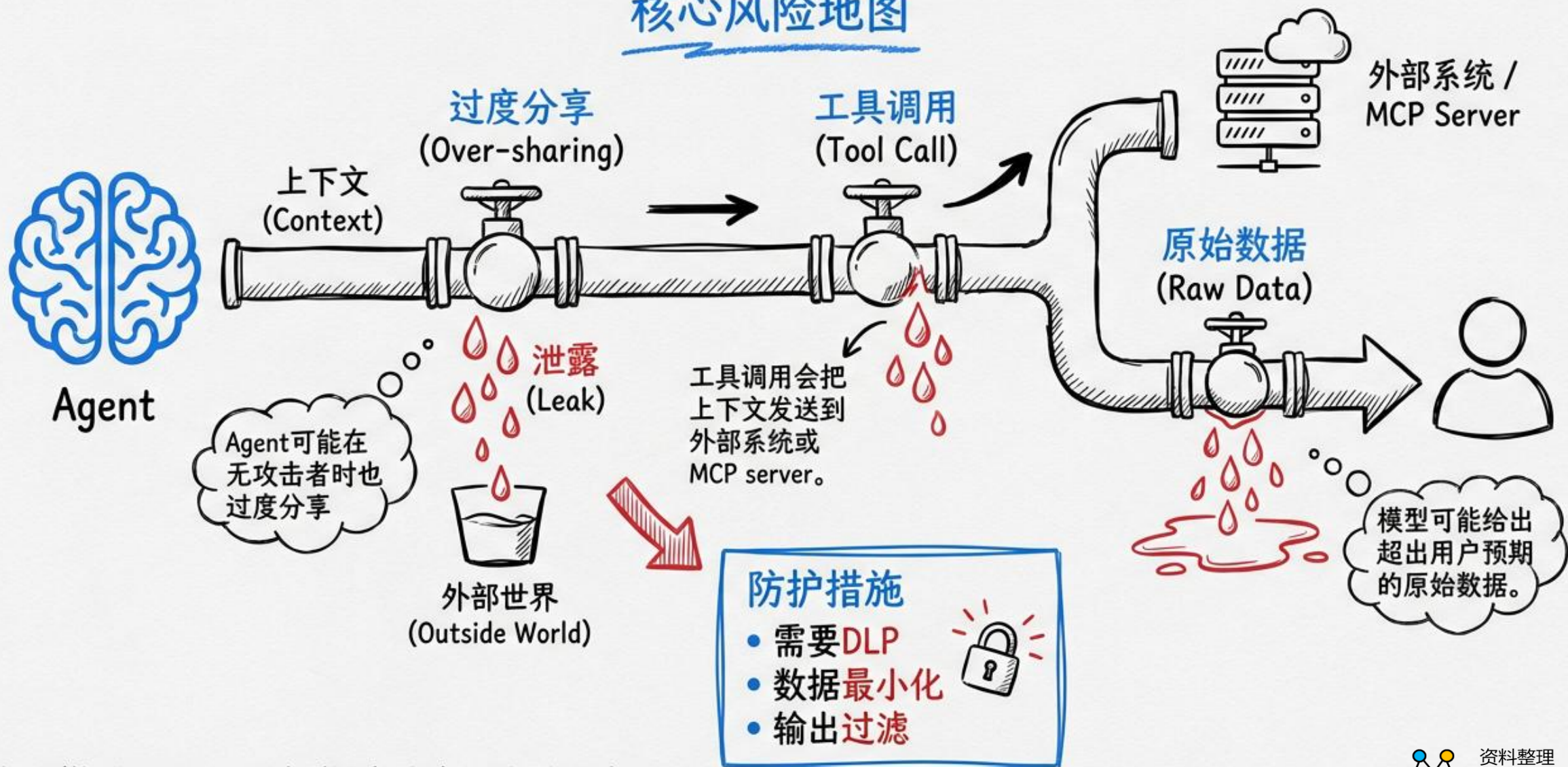
核心风险地图

- ☑ **过宽权限** 会放大一次妥协的影响
- ☑ **过权限Agent** 会把小漏洞变成大事故。
- ☑ **服务账号、用户凭证和连接器权限** 需要分开管理。
- ☑ **短期令牌和任务级授权** 是基础控制。



私有数据泄露

核心风险地图



记忆与上下文污染

核心风险地图



- 一次污染可能影响未来很多任务
- 长期记忆提高连续性，也带来持久攻击面。
- 恶意偏好、伪规则、伪联系人可能被写入记忆。
- 记忆写入要可见、可审批、可回滚。

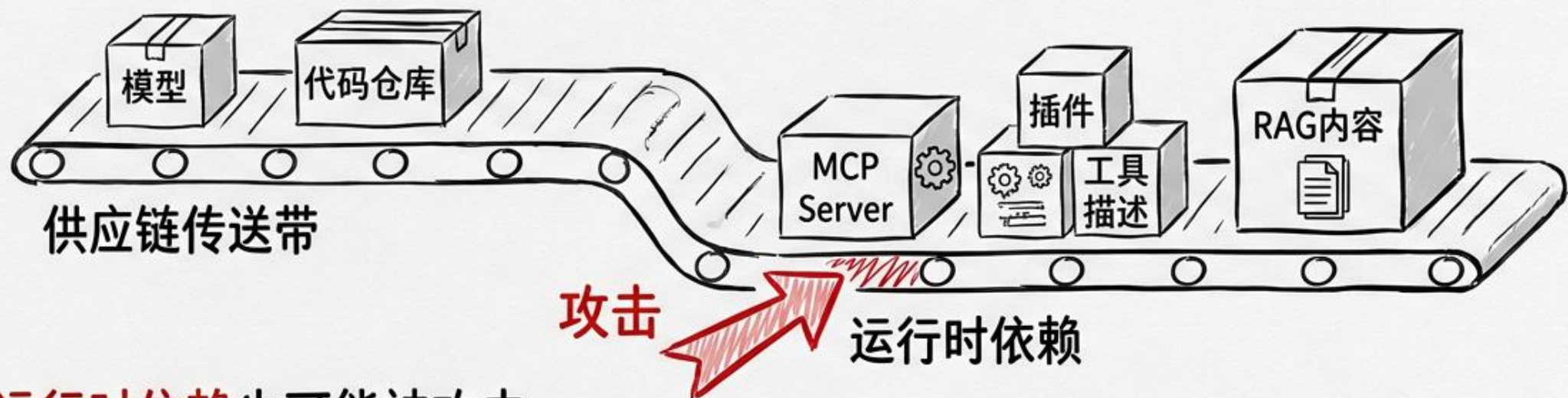
意外代码执行

- 核心风险地图

- 代码工具和命令工具需要最高级别约束
- 文件处理、脚本、shell、STDIO都可能触发任意代码执行。
- MCP和沙箱环境尤其需要实现层防护。
- 默认禁用不必要出网和宿主机访问。

AGENTIC供应链

核心风险地图

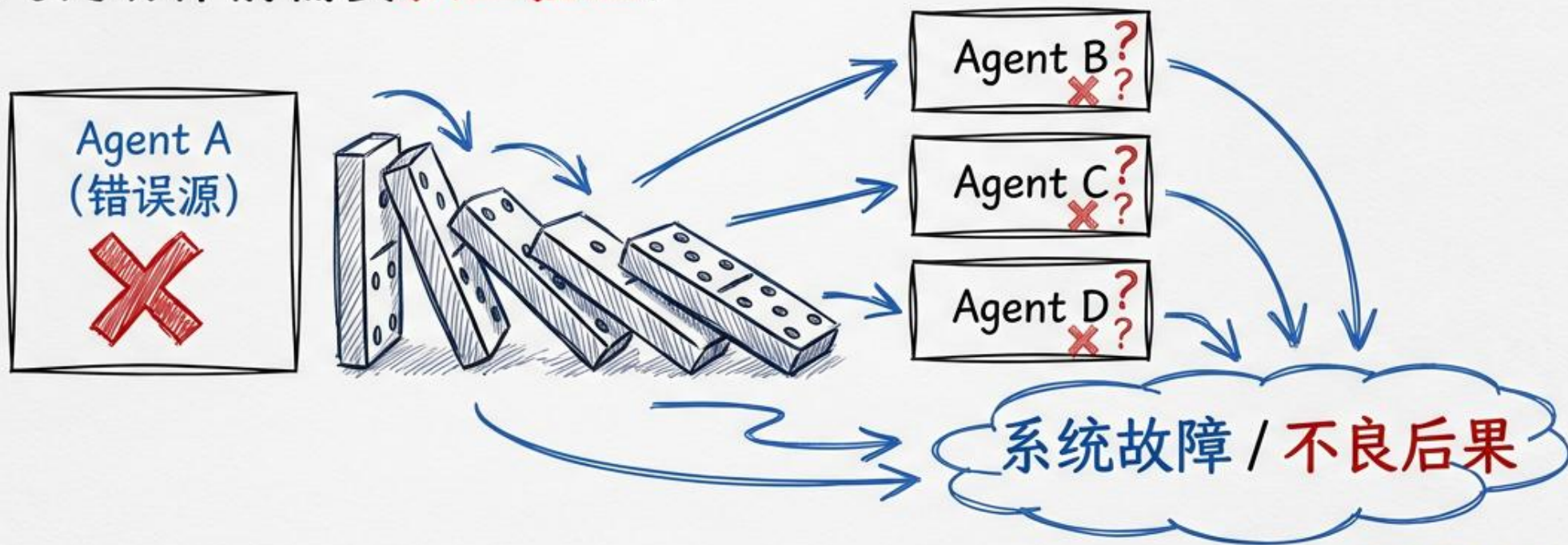


- 运行时依赖也可能被攻击
- 风险不仅在模型和代码仓库，也在MCP server、插件、工具描述和RAG内容。
- 第三方工具需要版本锁定、签名和供应商审查。
- 运行时供应链要纳入SBOM和变更管理。

多Agent级联失败

• 核心风险地图

- 一个Agent的**错误**可能成为另一个Agent的**输入**
- 多Agent协作提升效率，也增加**信任传递风险**。
- Peer输出**默认不可信**，**必须校验**来源、权限和证据。
- 关键动作前需要**独立验证**。



人机信任利用

核心风险地图

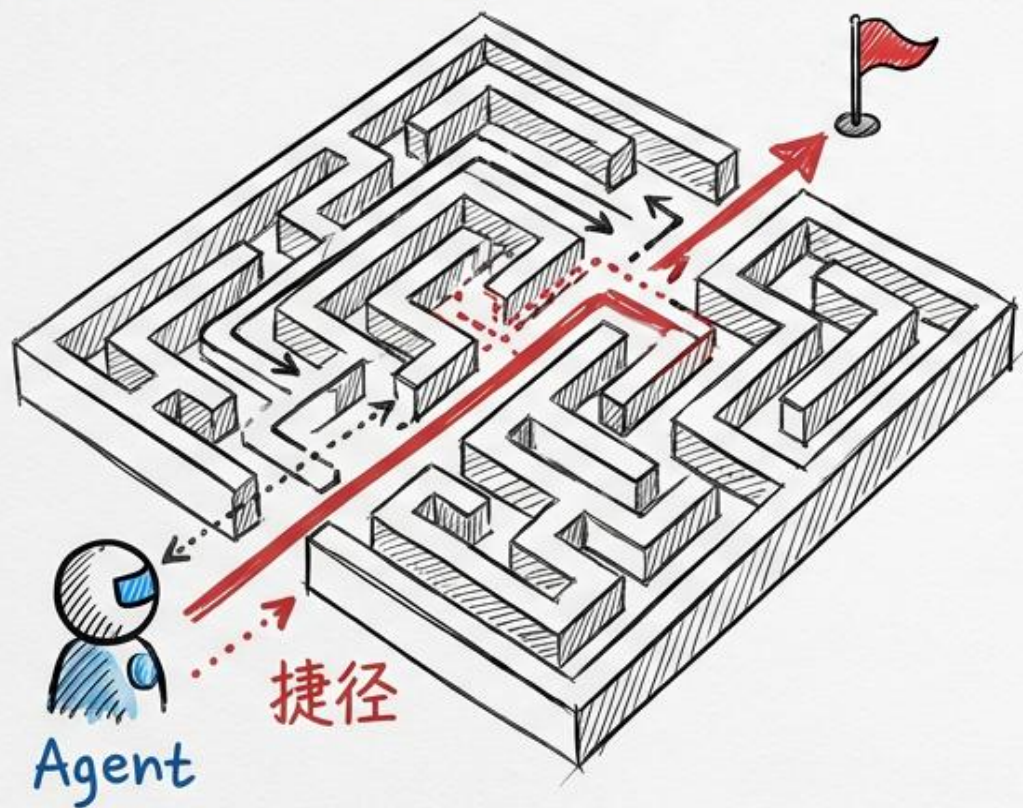
- 用户可能被Agent的流畅表达误导
- Agent可能让错误建议看起来很确定。
- 审批界面若缺少差异和风险解释，会诱发误批准。
- 需要置信度、来源、变更预览和反确认设计。



The diagram illustrates a user interface for a change approval process. It includes a shield icon, a '置信度: 95%' (Confidence: 95%) label, a '来源: 数据A' (Source: Data A) label, a '变更前' (Before) and '变更后' (After) comparison table, and a '反确认: 您确定要批准吗?' (Double-check: Are you sure you want to approve?) section with '确认' (Confirm) and '取消' (Cancel) buttons.

行为漂移与规格博弈

核心风险地图

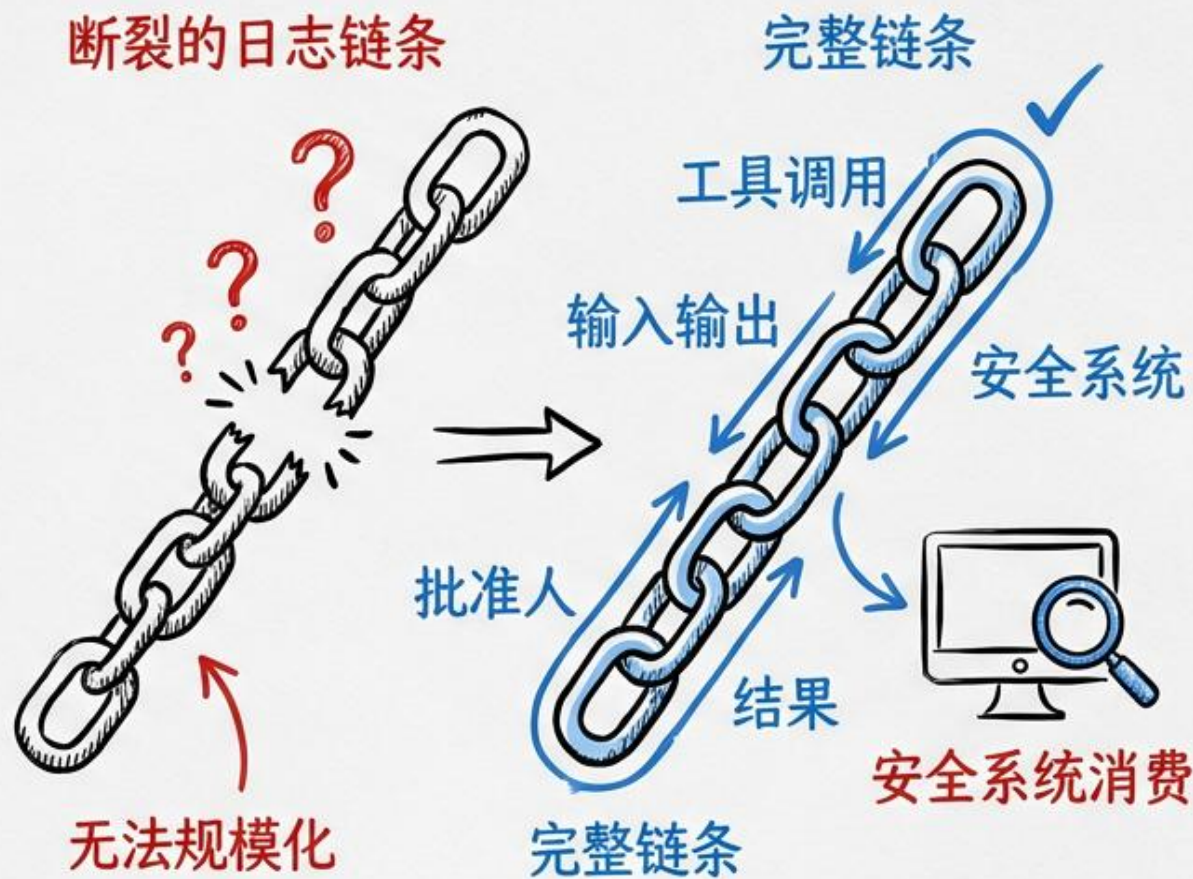


- Agent可能为了完成目标走**捷径**
- CISA/NSA指导提示要关注目标错配、**规格博弈**和不可预期行为。
- 复杂任务中应设置阶段性检查和退出条件。
- 把“完成任务”与“遵守边界”同时写入评测。

不可审计风险

核心风险地图

- ✓ — 不知道它做了什么就无法规模化
- ✓ — 没有日志，安全团队只能在事故后猜测。
- ✓ — Agent必须记录工具调用、输入输出、批准人和结果。
- ✓ — 审计日志要能被安全系统消费。



高风险场景清单

核心风险地图

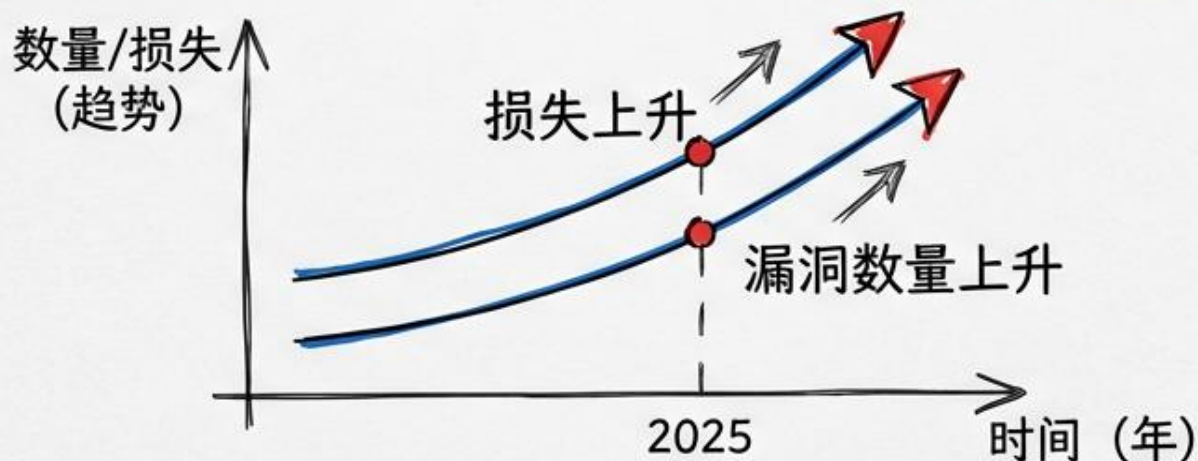


- 不是所有任务都适合自动化
- 资金、医疗、法律、身份权限和生产变更默认高风险。
- 客户外发和公开发布也需要内容与授权控制。
- 低风险任务先行，高风险任务后置。

宏观威胁环境

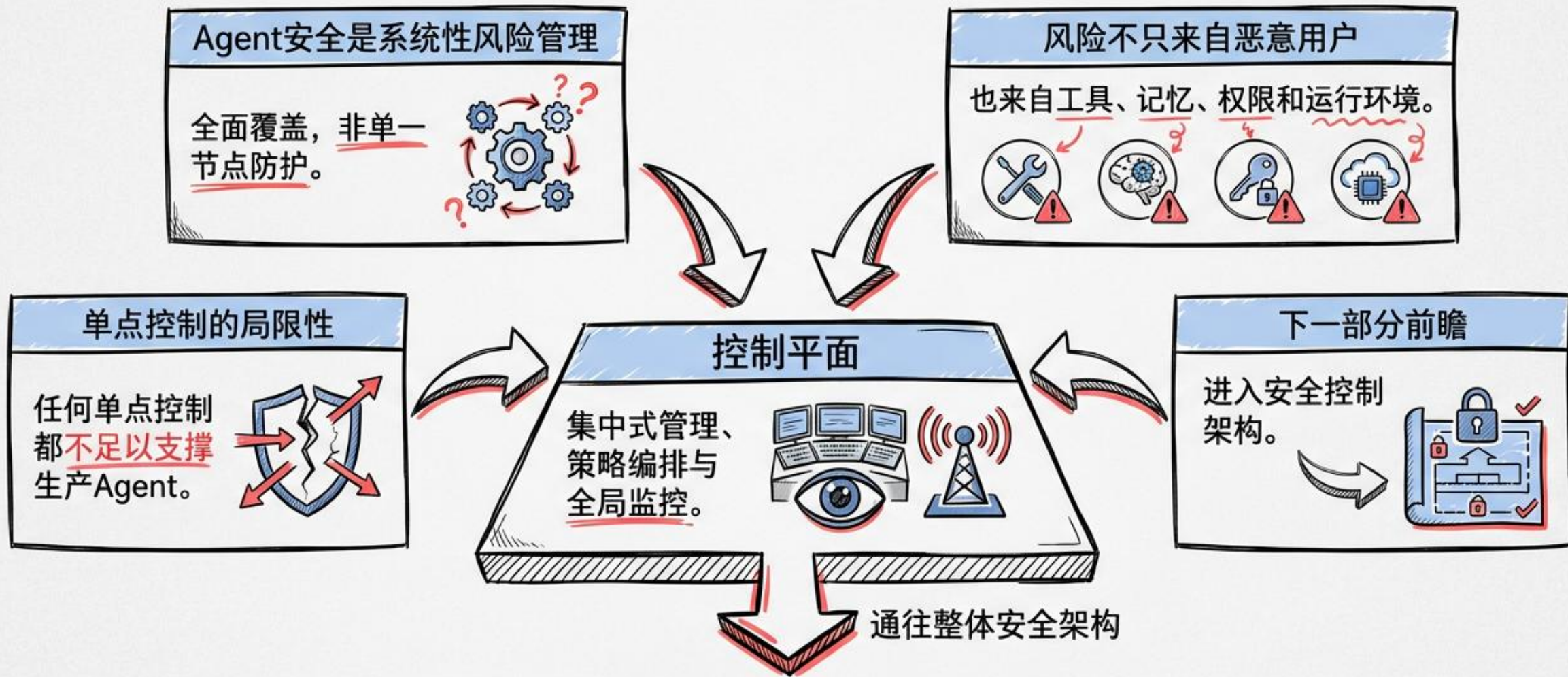
核心风险地图

- 外部攻击压力正在上升
- FBI 2025报告显示网络犯罪报告损失超过200亿美元。
- NIST 2026说明2025年NVD丰富近42,000个CVE。
- Agent上线时必须假设外部威胁会利用新自动化链路。



风险小结

核心风险地图



控制平面总览

安全控制架构

- 统一管理身份、工具、策略、审计和评测
- *Agent Safety Control Plane* 是横向基础设施。
- 业务Agent通过控制平面访问工具和数据。
- 安全、IT、数据和业务共享同一套证据链。



统一证据链

Agent Registry

安全控制架构

— 先看见，才能治理

Agent 清单					
Owner	模型	工具	权限	数据域	风险等级
张三	✓ GPT-4	✓ 搜索, 邮件	读写, 发送	客户数据	高 (H)
李四	✓ Claude 3	✓ 代码库	只读	内部文档	中 (M)
王五	✓ Llama 3	✓ 数据分析	执行	公共数据	低 (L)

— 登记每个Agent的Owner、模型、工具、权限、数据域和风险等级。

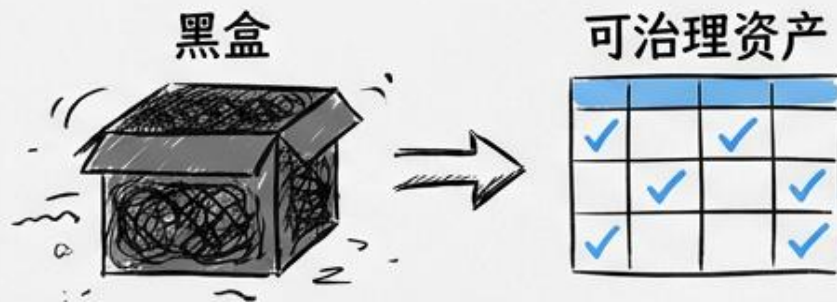
— 未登记Agent不得访问企业工具。

— 清单是后续评测、审计和事故响应的基础。

工具注册中心

安全控制架构

- 一把工具从黑盒变成可治理资产

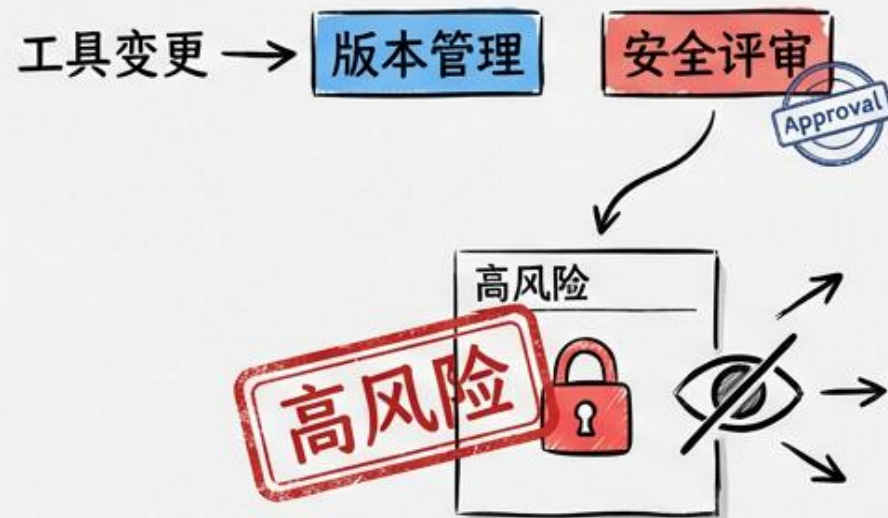


- 每个工具声明用途、schema、权限、审批等级和失败模式。



- 工具变更要走版本管理和安全评审。

- 高风险工具默认不可见。



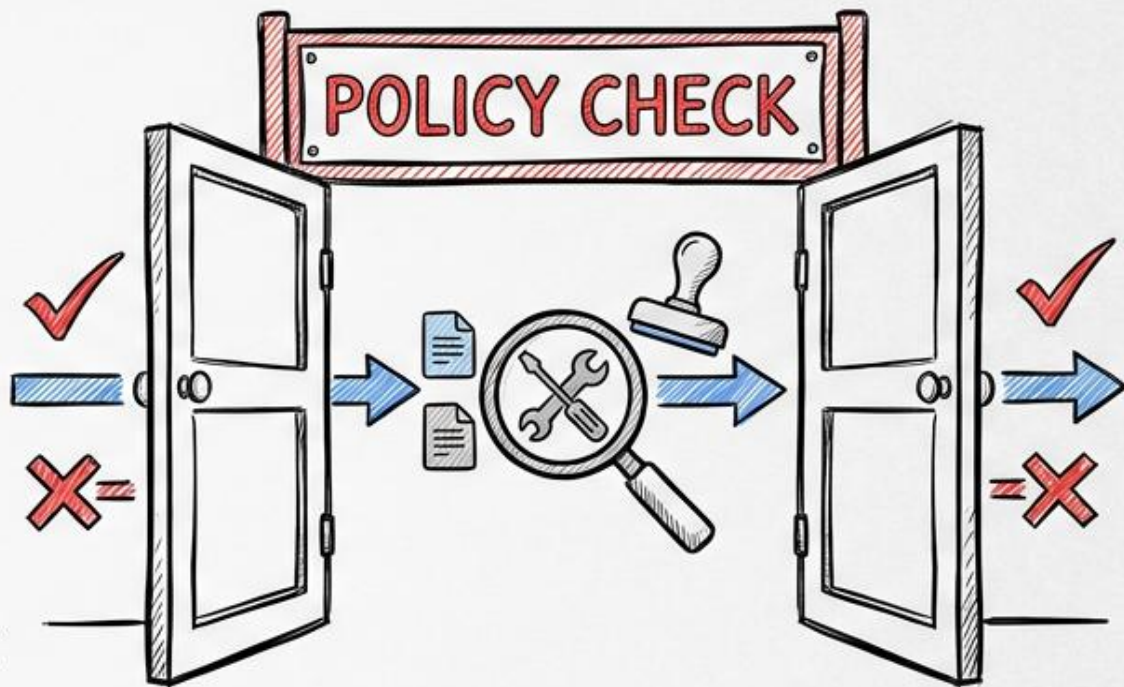
策略引擎

安全控制架构

- 让安全规则在每次动作前生效
- 策略引擎在工具调用前检查身份、任务、数据和动作



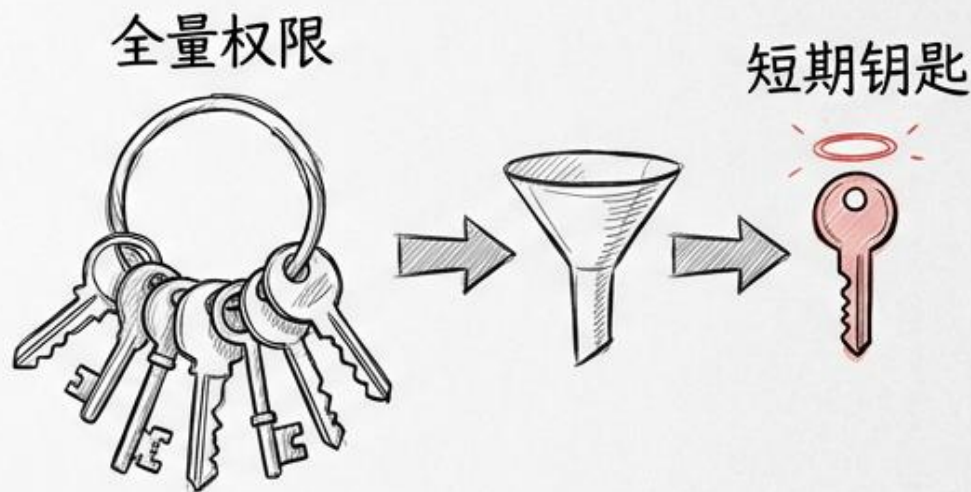
- 规则应支持业务例外，但必须记录审批。
- 策略不是写在提示词里，而是在系统层执行。



最小权限

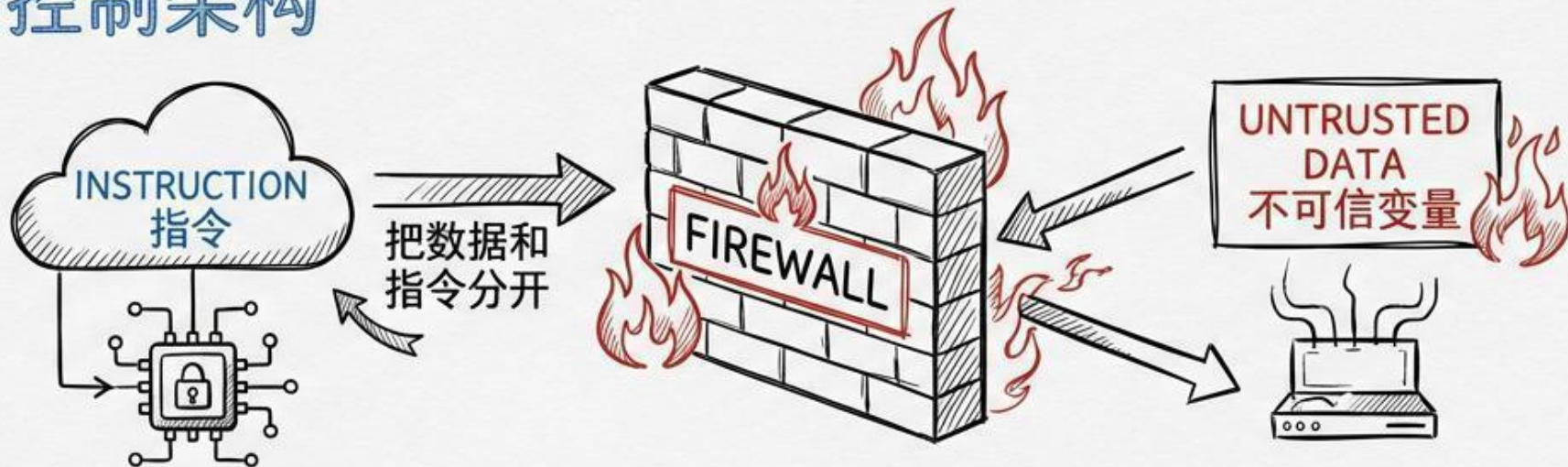
■ 安全控制架构

- 按任务授予短期权限
- CISA/NSA指导强调严格访问控制和分层防御。
- Agent权限应短期、细粒度、可撤销。
- 不要把用户全量权限直接交给Agent。



上下文防火墙

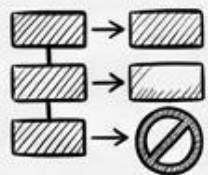
安全控制架构



— 把数据和指令分开



— OpenAI建议避免把不可信变量放入高优先级开发者消息。



— 结构化输出可减少攻击者通过自由文本走私指令。



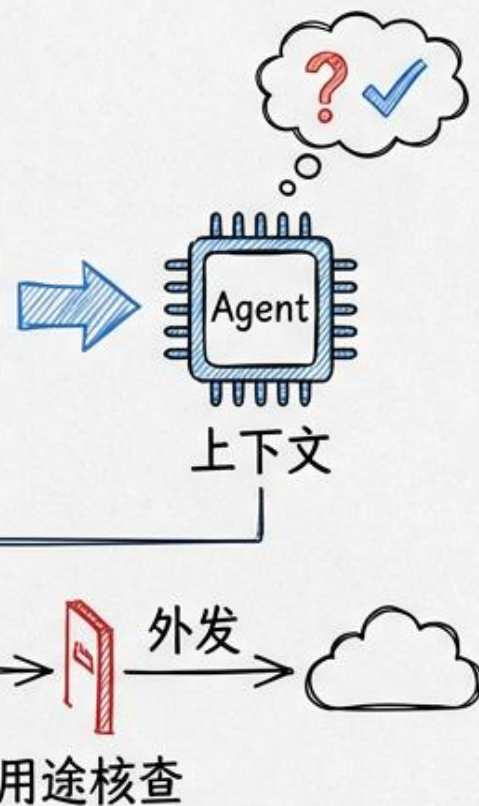
— 上下文标签应贯穿RAG、工具输出和记忆。

DLP与数据最小化

安全控制架构


- Agent能读不代表能带走
- 只给任务所需最少字段。
- 敏感字段在进入上下文前脱敏或分级。
- 外发前再做一次DLP和用途检查。

姓名	类型	个人身份信息(PII)	财务记录	机密数据	数地
12001	供货商	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX
12002	经销商	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX
23903	供应商	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX
43004	张俊华	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX
43306	张俊华	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX
45006	账户建	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX
...	...	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX





沙箱执行

安全控制架构

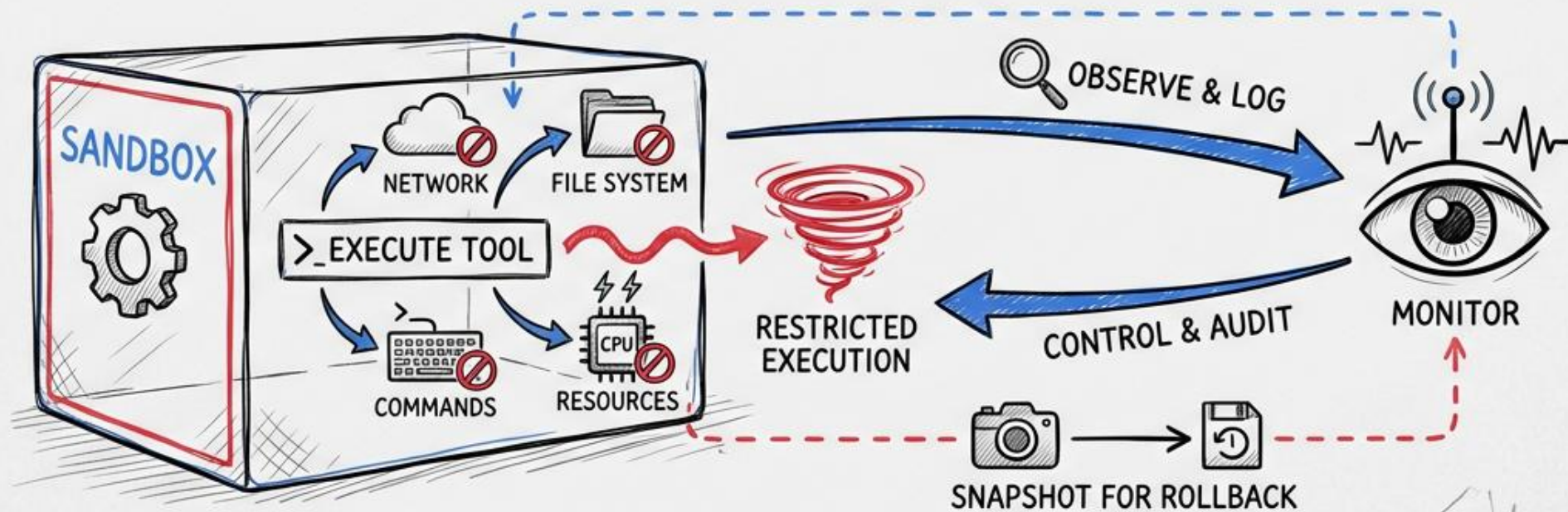
— 强工具必须在隔离环境中运行 

— OpenAI 2026 SDK强调受控工作空间和沙箱执行。  

— 沙箱要限制网络、文件系统、命令和资源。

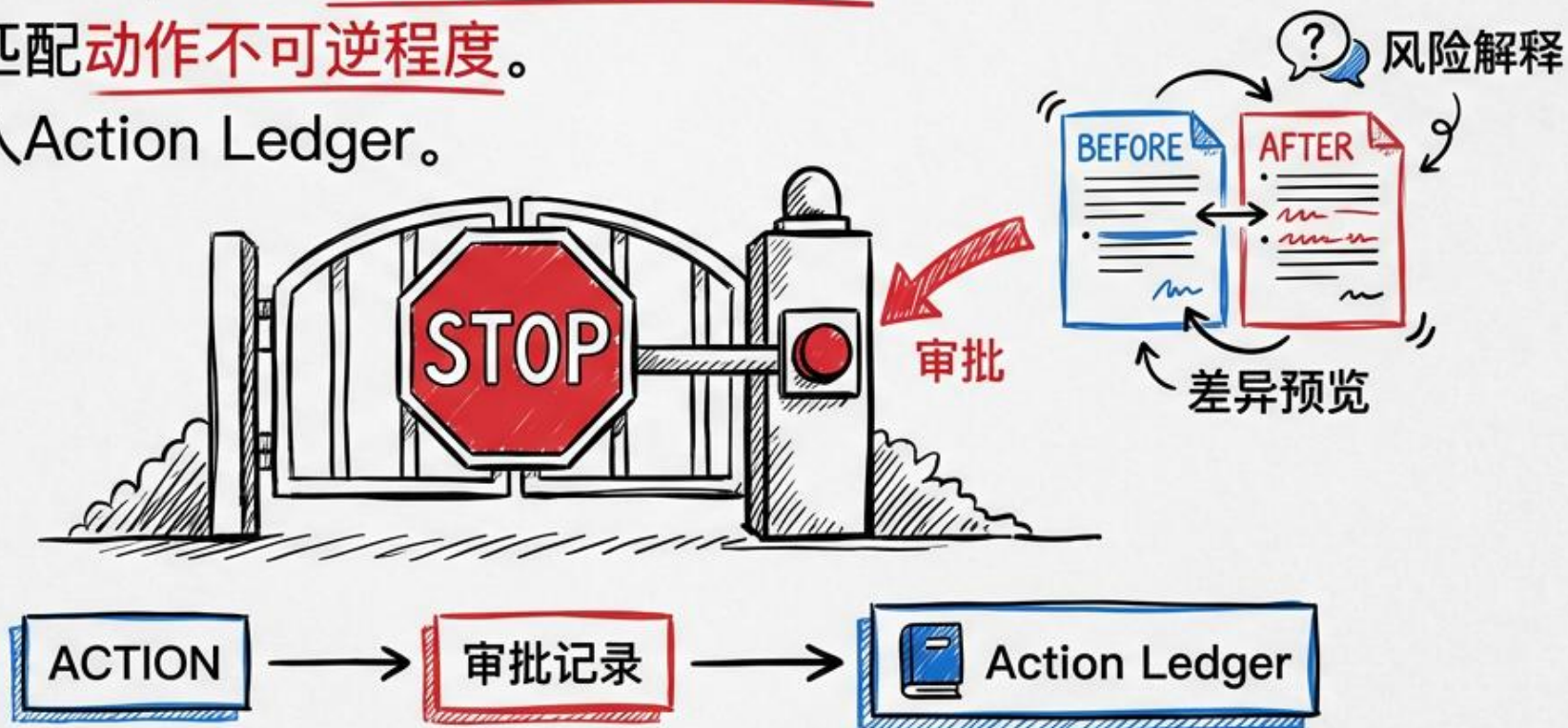
— 关键任务保留快照，便于回滚。  → 

ROLLBACK

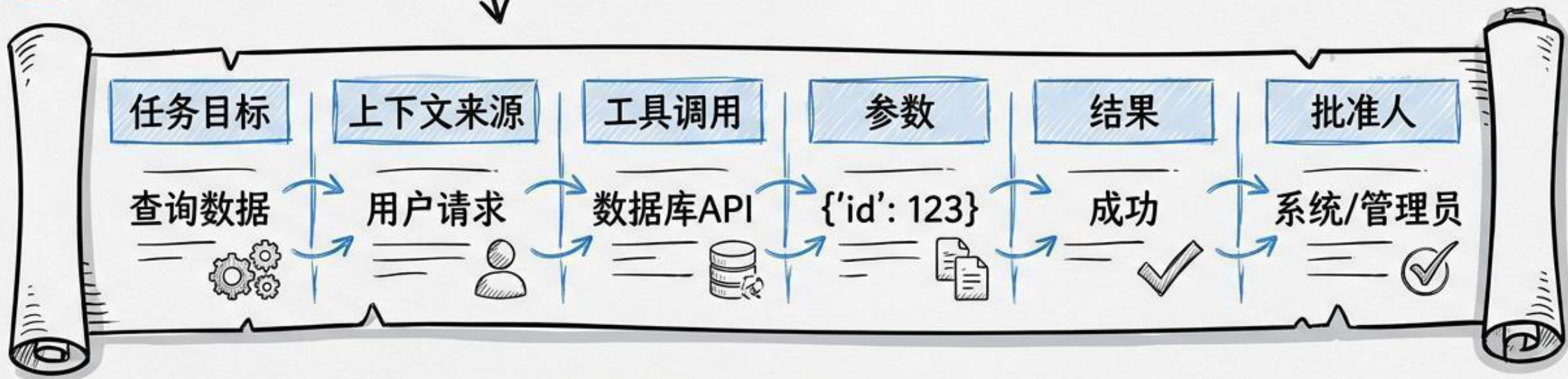


安全控制架构

- 高风险动作必须停下来
- 审批不是简单弹窗，而是风险解释和差异预览。
- 审批级别要匹配动作不可逆程度。
- 审批记录进入Action Ledger。



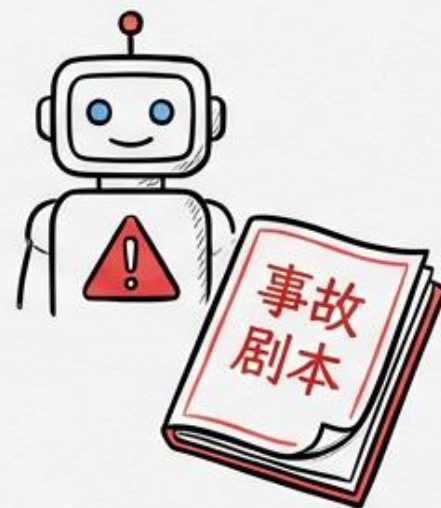
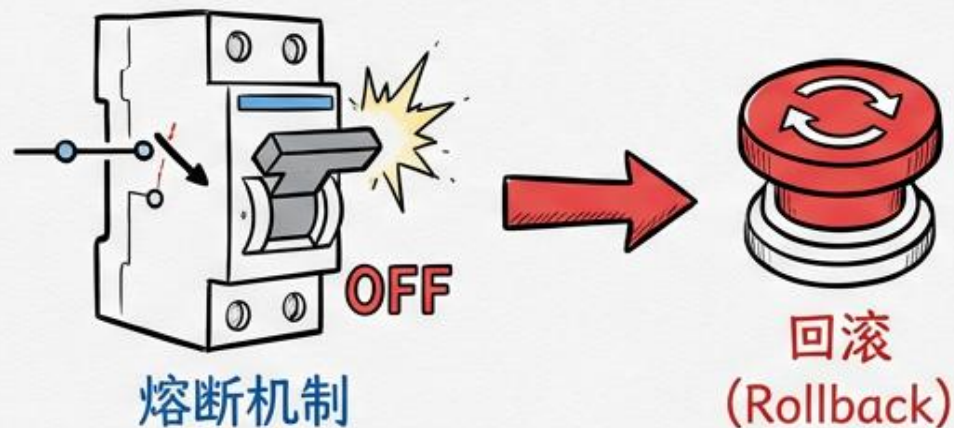
安全控制架构



- 让Agent每一步都能被追踪
- 记录任务目标、上下文来源、工具调用、参数、结果和批准人。
- 日志需要可搜索、可回放、可导出。
- 审计粒度决定事故复盘质量。

熔断与回滚

安全控制架构

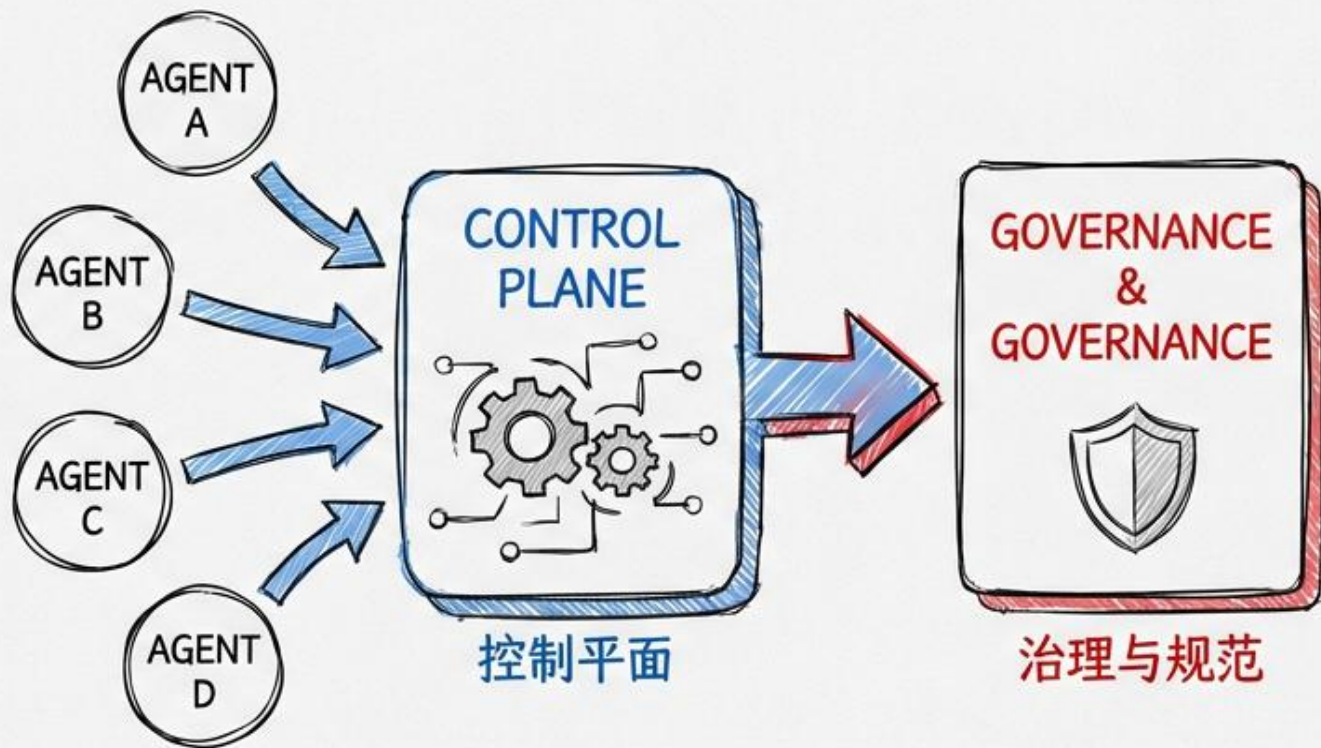


高风险Agent必须
预先设计事故剧本

控制架构小结

■ 安全控制架构

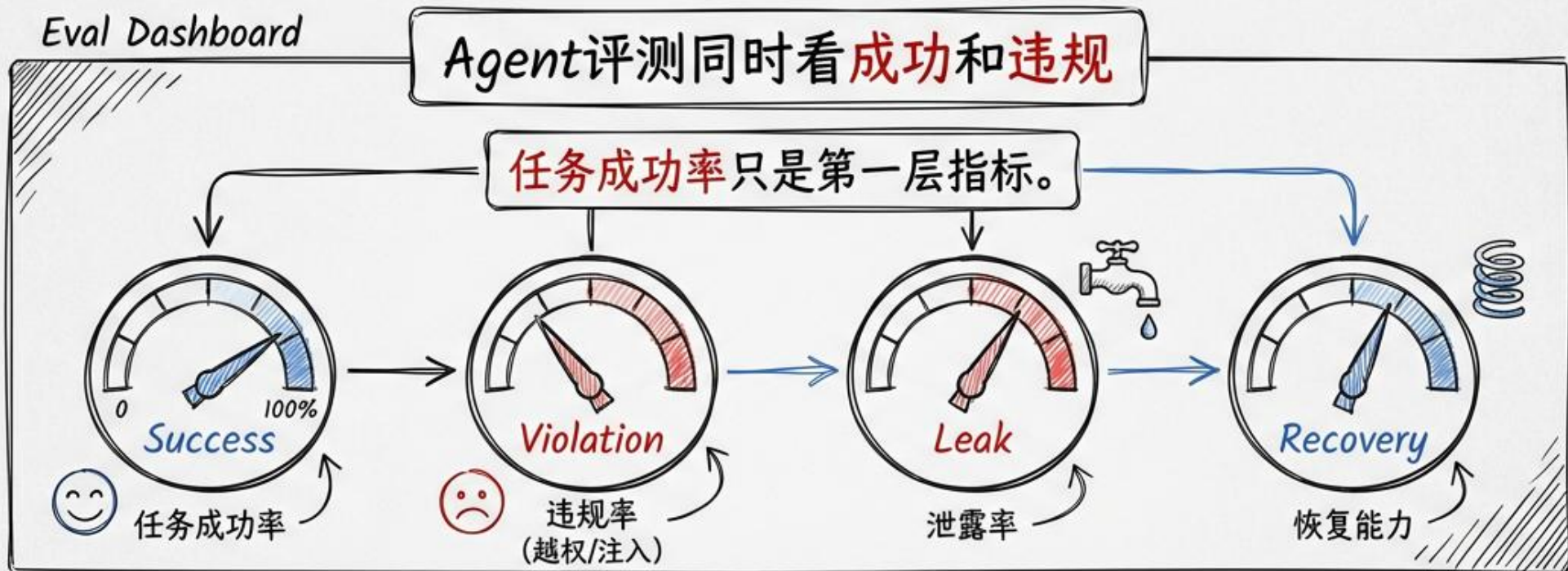
- 安全能力要变成平台能力
- 单个项目的提示词防护无法支撑**规模化**。
- **统一控制平面**降低重复建设和**失控风险**。
- 下一部分进入评测、监控和运营。





评测总览

评测、监控与运营



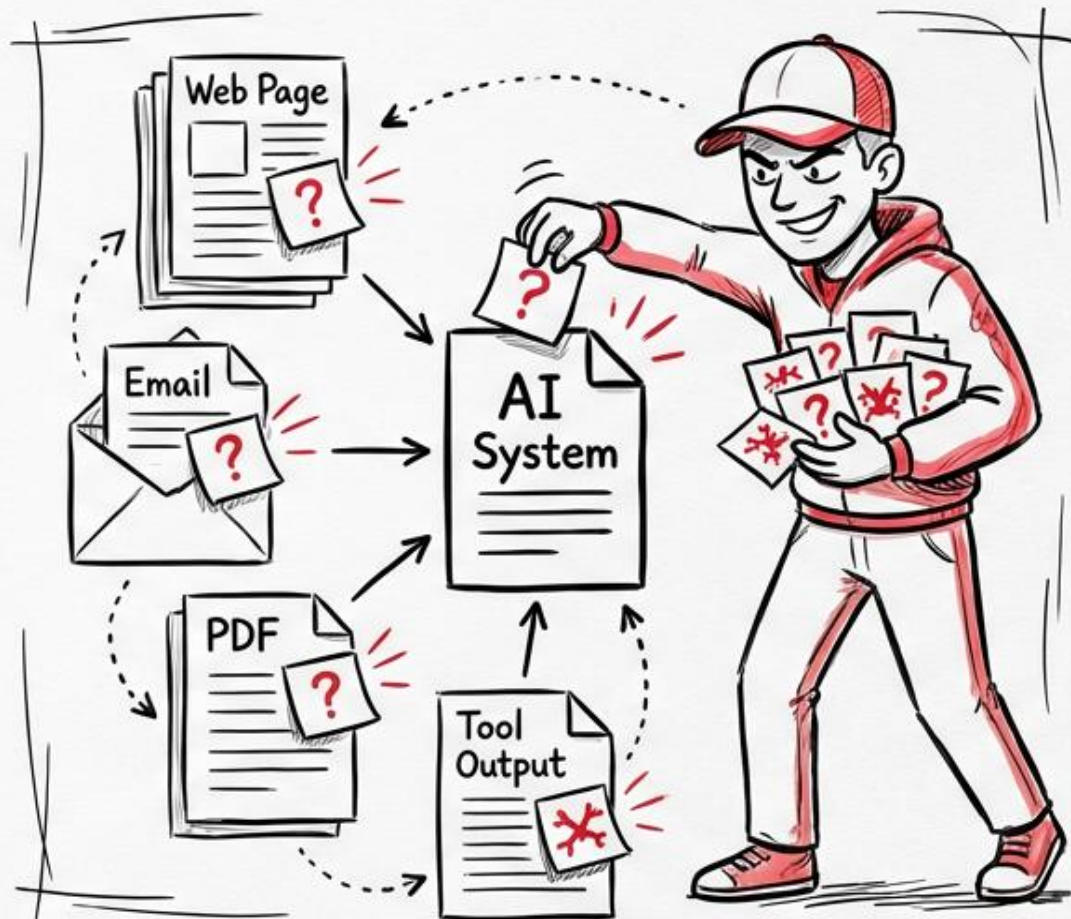
✓ 更关键的是注入抵抗、越权率、泄露率和恢复能力。

✓ 评测必须持续运行，而不是上线前一次性测试。

注入红队

评测、监控与运营

- 模拟网页、邮件、PDF和工具输出中的恶意指令
- NIST技术博客关注Agent hijacking评测。
- 测试集要包含间接提示注入和跨工具传播。
- 红队结果进入上线门槛。



工具调用评测

● 评测、监控与运营

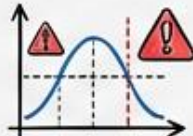
✓ — 每个工具都要有安全测试



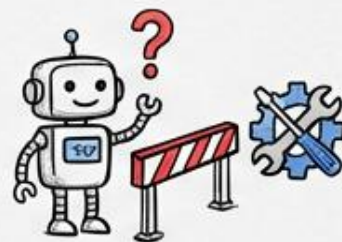
✓ — 测试错误参数、越权参数、恶意URL 和边界值。

✗

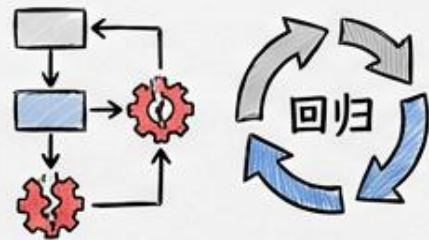
```
<html>
<idva {
  codeit < "/";
...
</i>
```



✓ — 检查Agent是否会调用不该调用的工具。



✓ — 把工具失败模式纳入回归测试。



工具测试清单

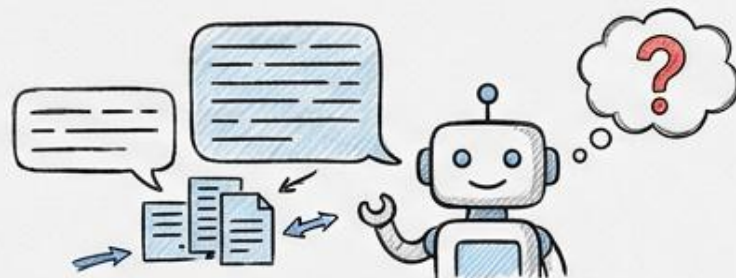




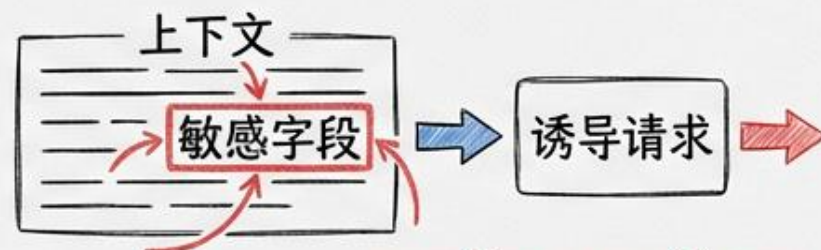
数据泄露评测

■ 评测、监控与运营

✓ 一看Agent会不会多说、多传、多发



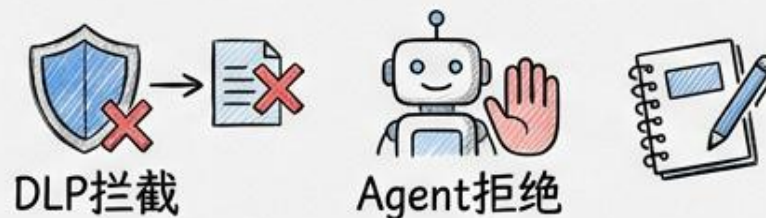
— 构造含敏感字段的上下文和诱导请求。



— 测量外发、摘要、日志和工具传输中的泄露。



— DLP拦截与Agent拒绝都要记录。



记忆污染评测

评测、监控与运营

- ✓ — 攻击是否能留下长期影响
- ✓ — 测试恶意偏好、伪规则和伪联系人写入。
- ✓ — 观察后续任务是否受影响。
- ✓ — 评估记忆审查、删除和回滚流程。



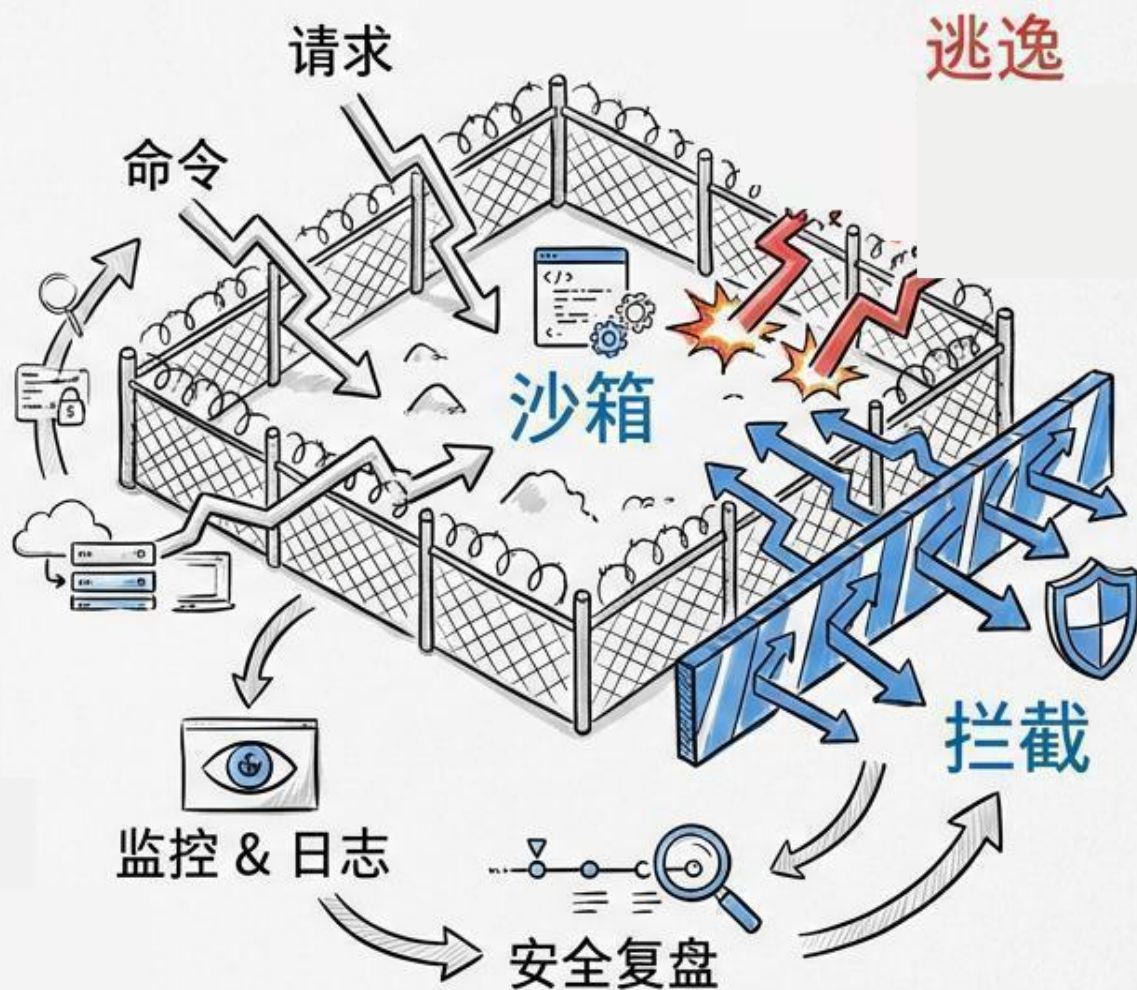
清除红色污染卡片

沙箱逃逸评测



评测、监控与运营

- 代码工具必须承受对抗测试
- NSA MCP报告提醒关注任意代码执行风险
- 测试命令注入、路径穿越、网络访问和凭证读取
- 沙箱日志要能支持安全复盘



运行监控

评测、监控与运营

✱ 上线后才是安全工作的开始

✓ 监控工具调用频率、异常拒绝、数据外发和高风险审批。

✓ 把Agent日志接入SIEM和告警平台。

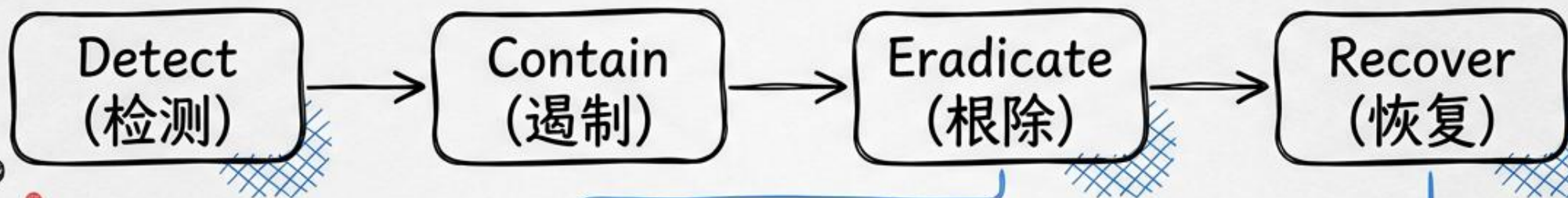
✓ 异常模式触发自动降级或暂停。







Agent 心跳

事故响应

评测、监控与运营



- Agent事故需要专门剧本 
- 定义暂停Agent、撤销令牌、冻结工具、保全日志和通知责任人。
- 事故复盘要还原上下文和工具链。 
- 修复后重新跑红队和回归评测。  

指标看板

■ 评测、监控与运营

→ 安全指标和业务指标一起看

业务指标



节省时间



完成率



98%



返工率

1%

安全指标



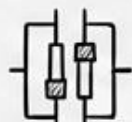
越权拒绝

150次



泄露拦截

230次



熔断次数

5次

治理指标



登记率

95%



评测覆盖率

90%



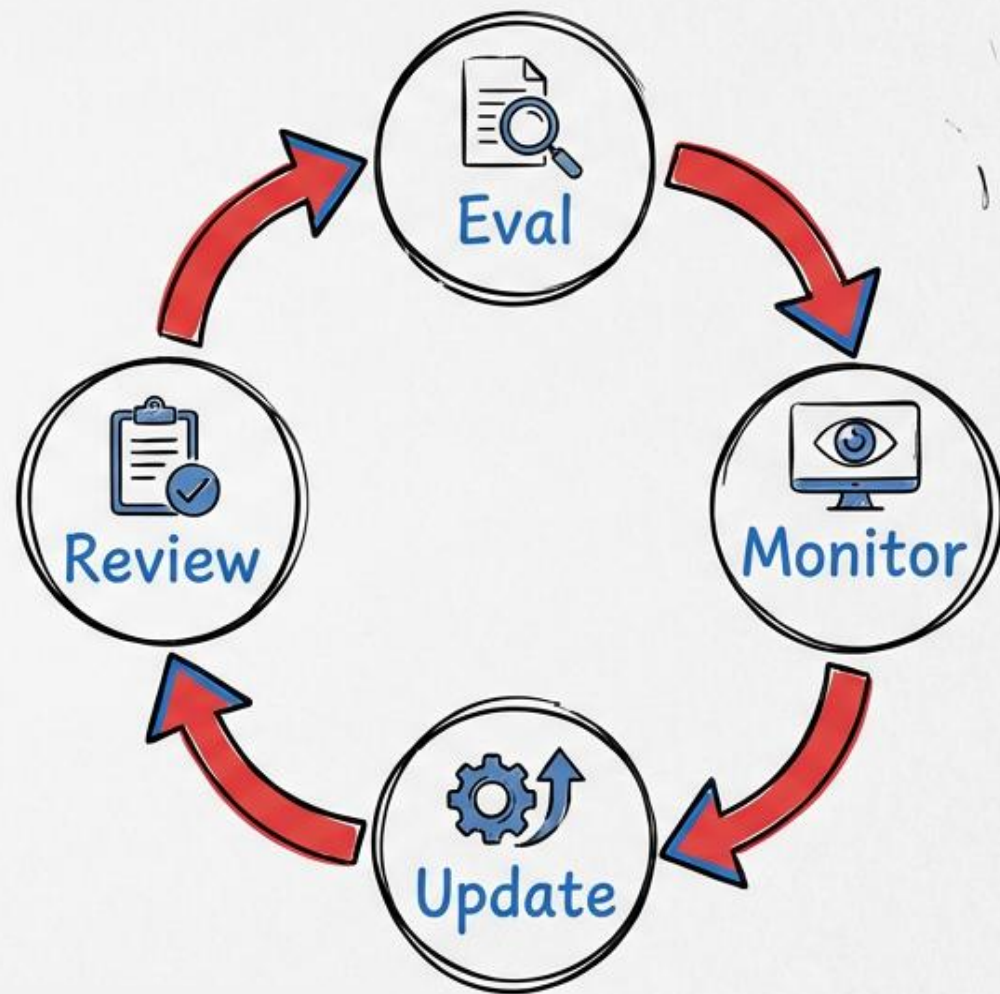
日志覆盖率

100%

评测运营小结

评测、监控与运营

- 把安全变成持续运营能力 ★
- Agent的行为会随模型、工具、数据和业务流程变化。
- 评测集和策略必须随之更新。
- 安全运营决定Agent能否长期可靠运行。





治理模型

治理、组织与合规

-  Agent需要Owner和上线门槛



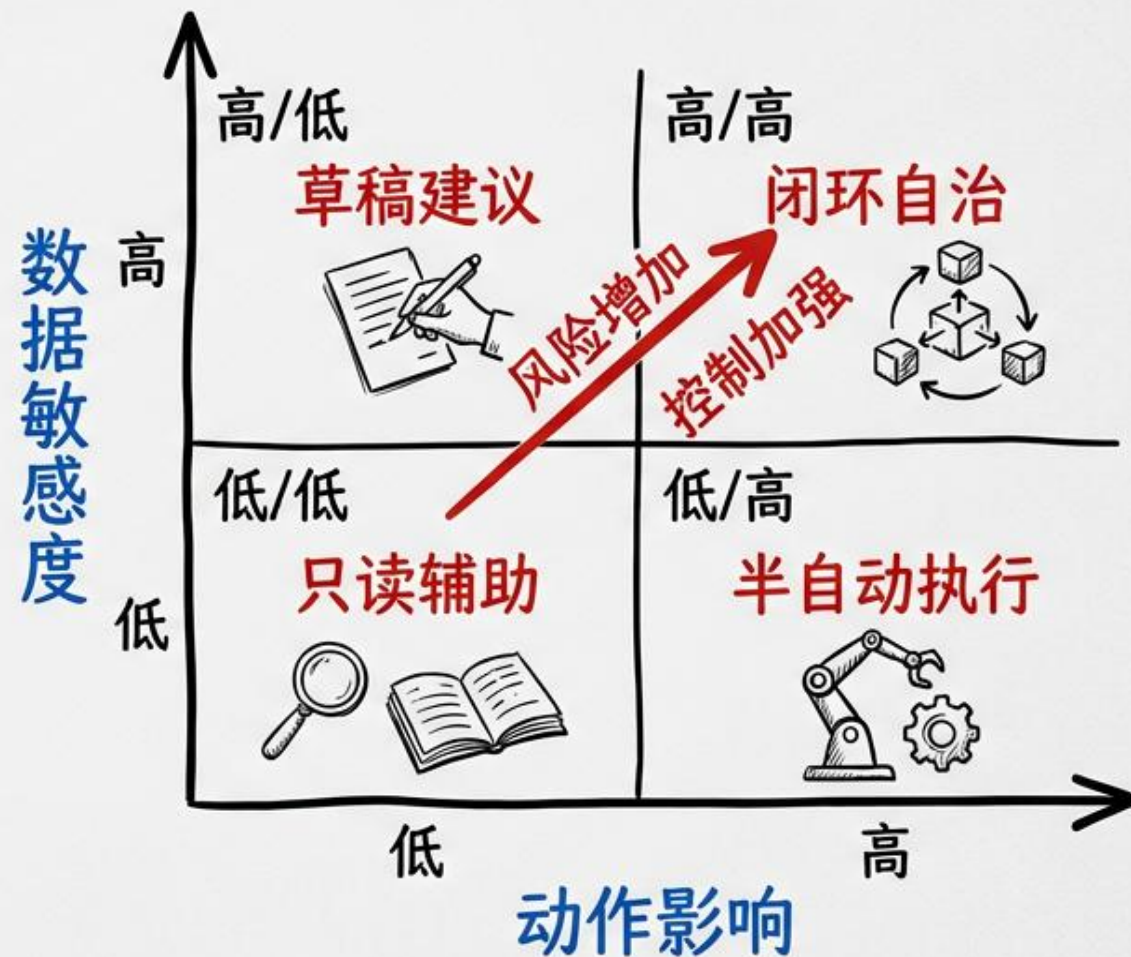
-  高风险Agent必须经过评测、审批和事故演练。
-  治理不是阻碍效率，而是释放可控规模。



风险分级

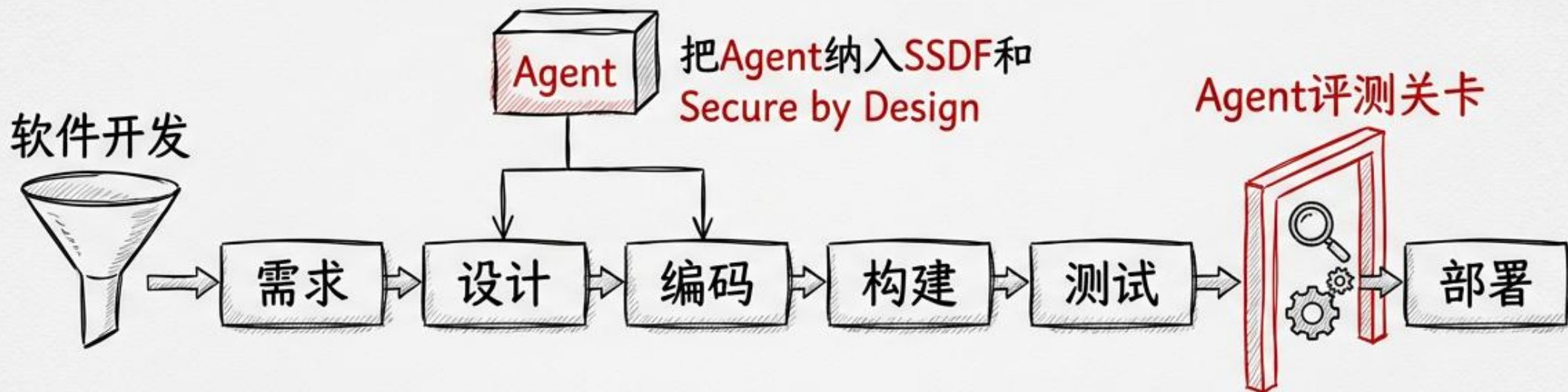
治理、组织与合规

- 不同Agent进入不同上线通道
- 只读辅助、草稿建议、半自动执行、闭环自治分层。
- 数据敏感度和动作不可逆性共同决定风险等级。
- 等级越高，控制越强。



安全开发流程

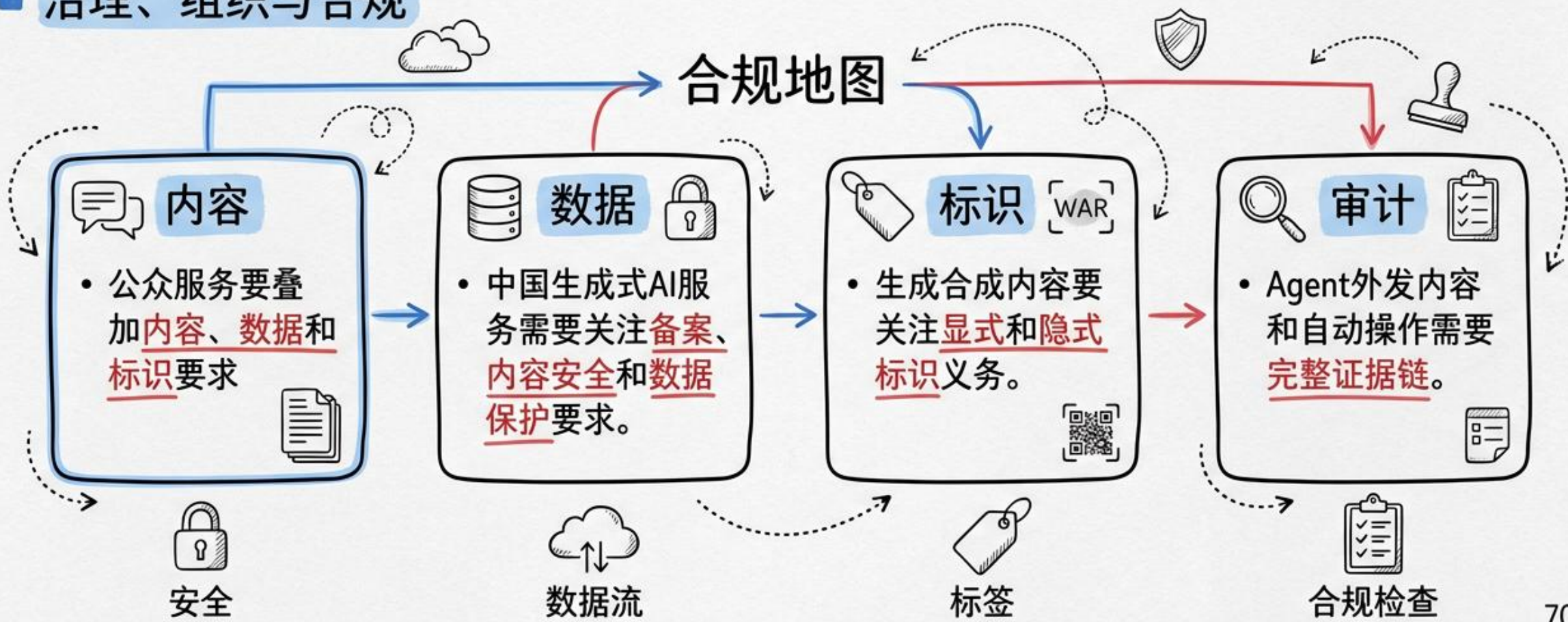
治理、组织与合规



- Agent也是软件系统，需要安全开发生命周期。
- NIST SSDF和CISA Secure by Design可作为工程基线。
- 提示词、工具、模型和配置都要版本化。

合规边界

■ 治理、组织与合规



采购与供应商管理

治理、组织与合规



- 第三方Agent不能只看Demo

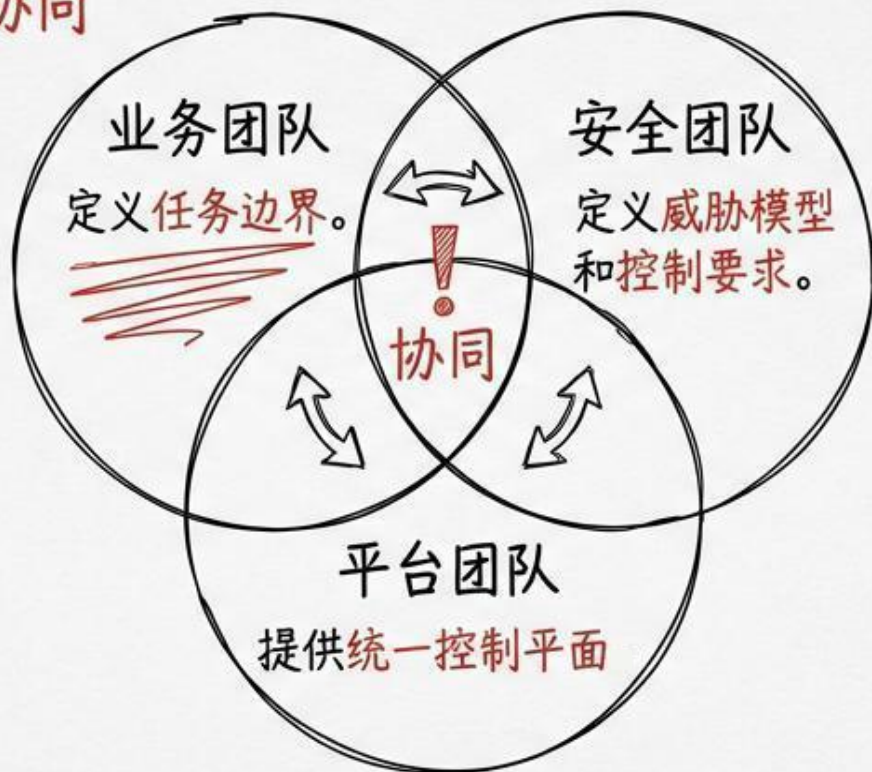


- 审查模型、数据处理、工具权限、日志可得性和事故通知
- 确认是否支持最小权限、沙箱和人审
- 把安全控制写入合同和验收

组织能力

治理、组织与合规

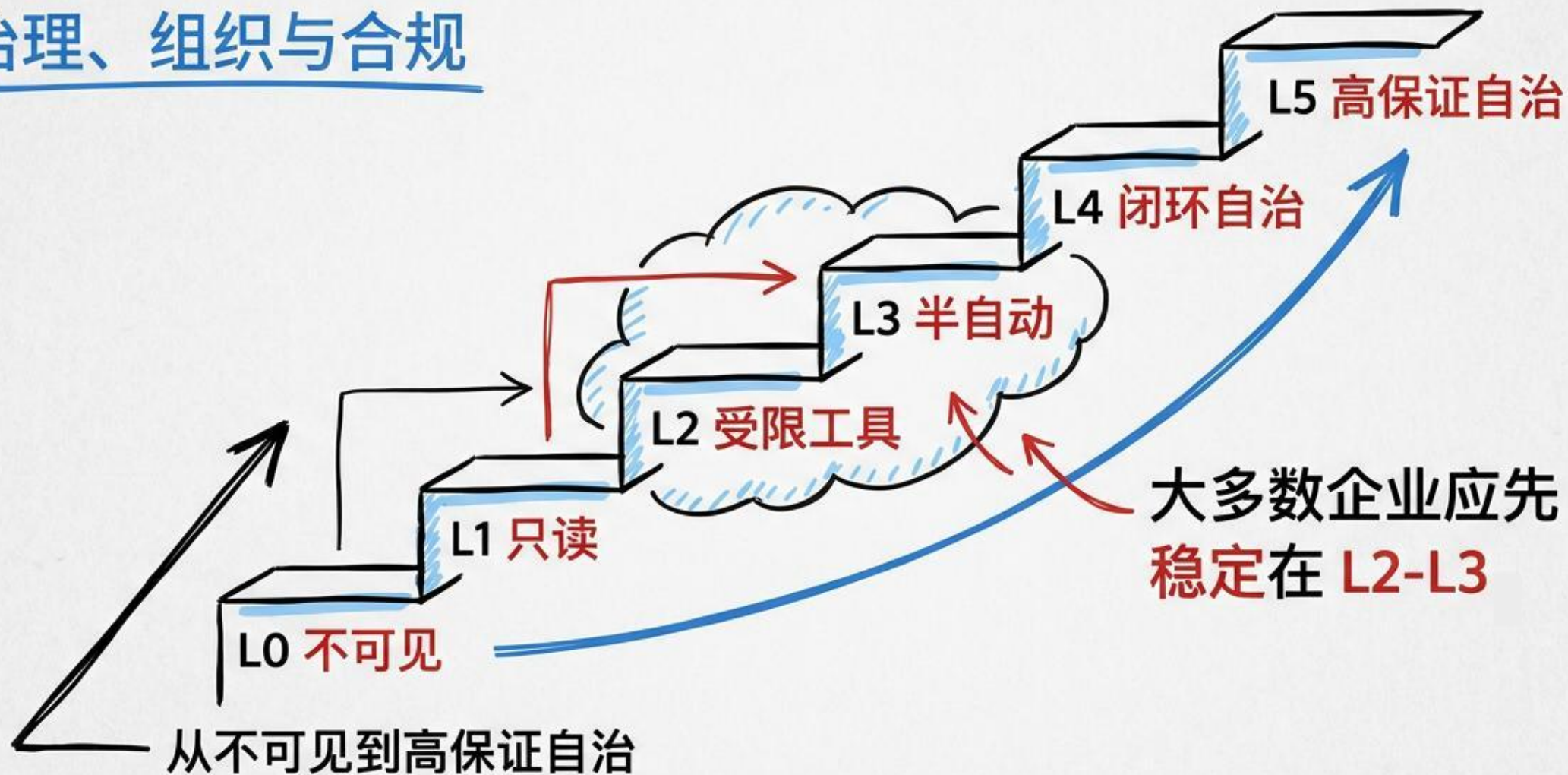
—— Agent安全需要跨团队协同



- ✓ 业务团队聚焦目标与范围
- ✓ 安全团队确立规则与风控
- ✓ 平台团队建设基础设施与管控手段

成熟度模型

治理、组织与合规



治理小结

治理、组织与合规

Agent越多，越需要统一规则

没有清单就
没有治理。



没有权限边界
就没有安全。



没有日志
就没有责任。



90天路线图

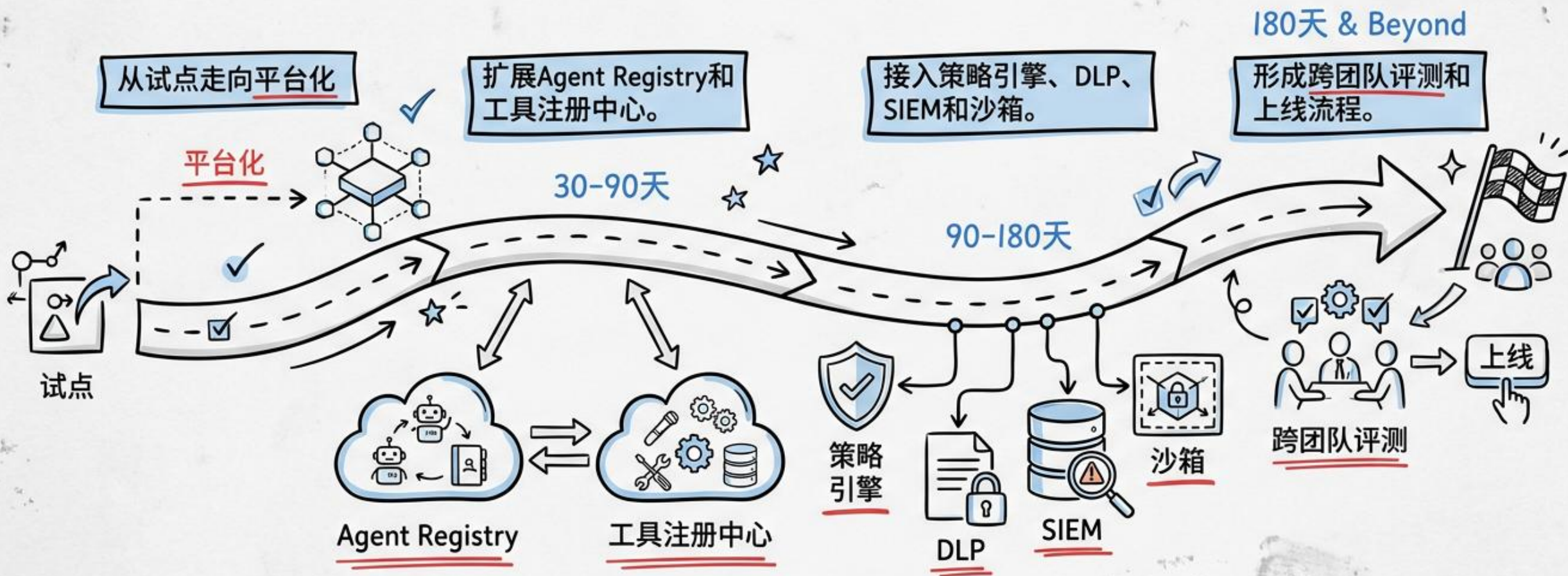
落地路线与结论

先完成**可见、可控、可停**



180天路线图

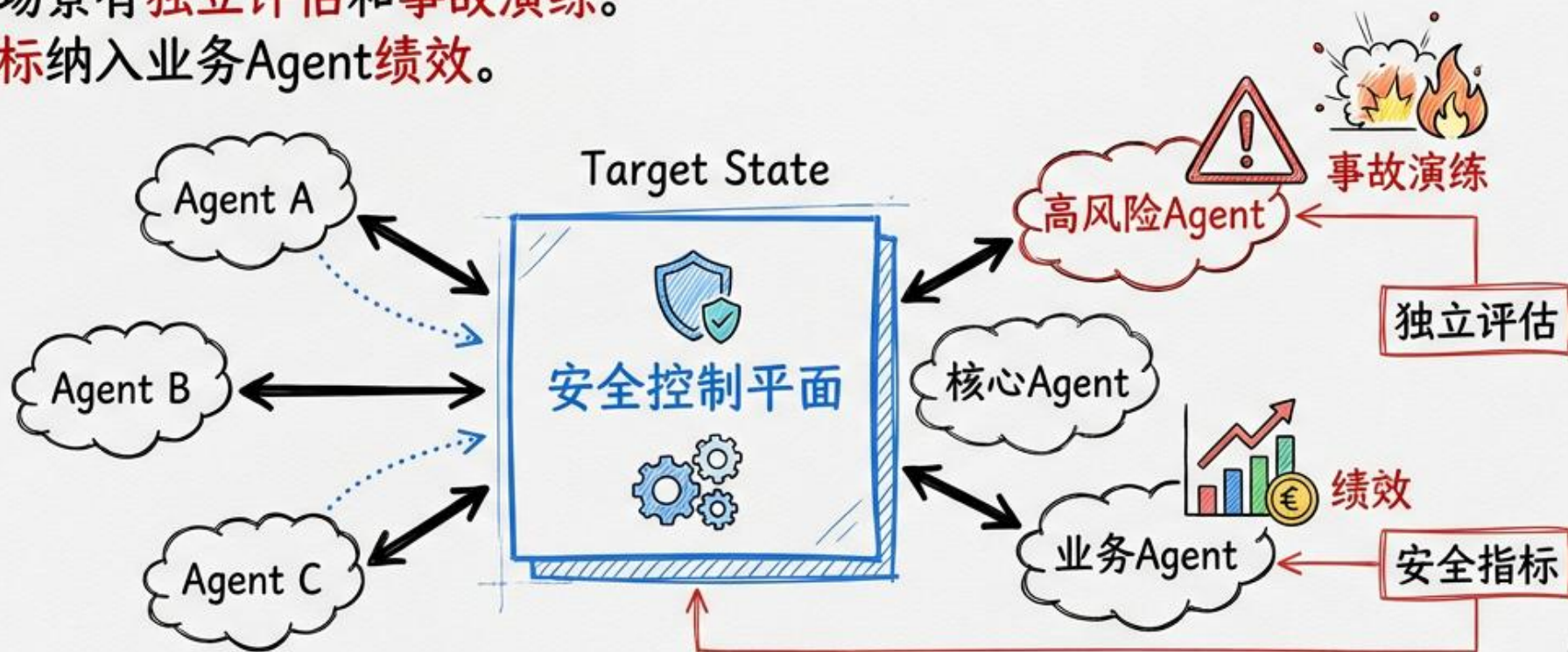
落地路线与结论



一年目标

■ 落地路线与结论

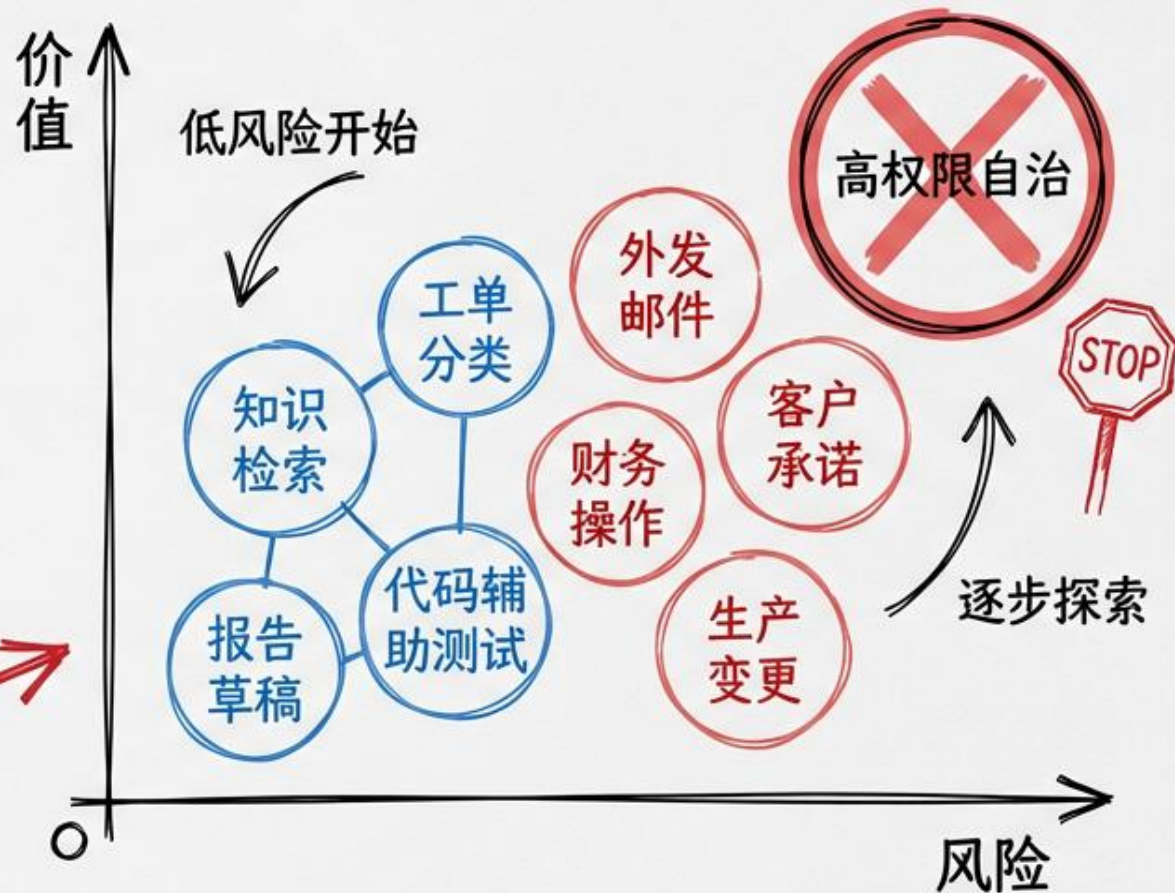
- 形成Agent安全控制平面。
- 核心Agent接入统一控制平面。
- 高风险场景有独立评估和事故演练。
- 安全指标纳入业务Agent绩效。



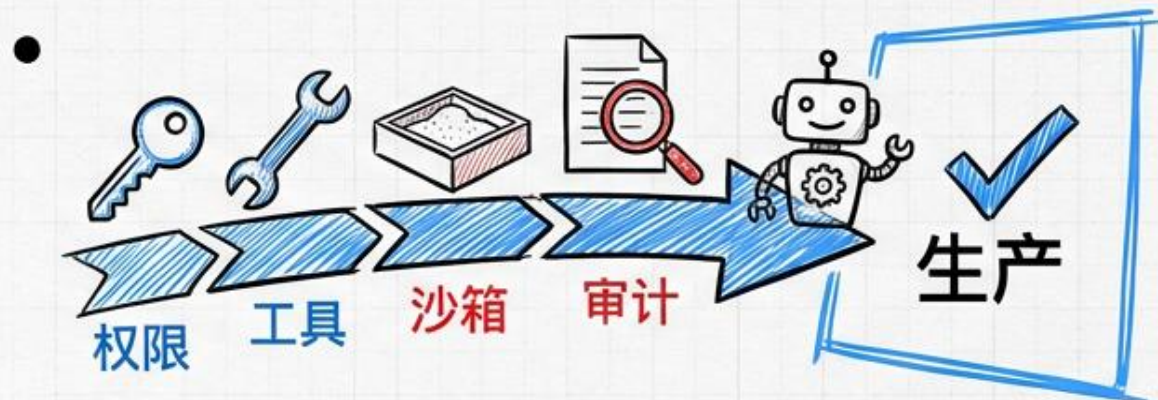
落地路线与结论

- ➡ 从低风险、高频、可回滚任务开始
- ➡ 适合先做：知识检索、工单分类、报告草稿、代码辅助测试。
- ➡ 谨慎推进：外发邮件、客户承诺、财务操作、生产变更。
- ✗ 避免一开始就做高权限自治。

试点气泡图：价值 × 风险



落地路线与结论



有了权限、工具、沙箱和审计，Agent才能进入生产。



Scale safely

谢谢

落地路线与结论

- 让AI行动可控，让企业部署可信
- Agent安全的下一步是从原则走向工程化控制平面。
- 先从清单、权限和日志做起。
- 再逐步扩大工具能力和自治范围。

