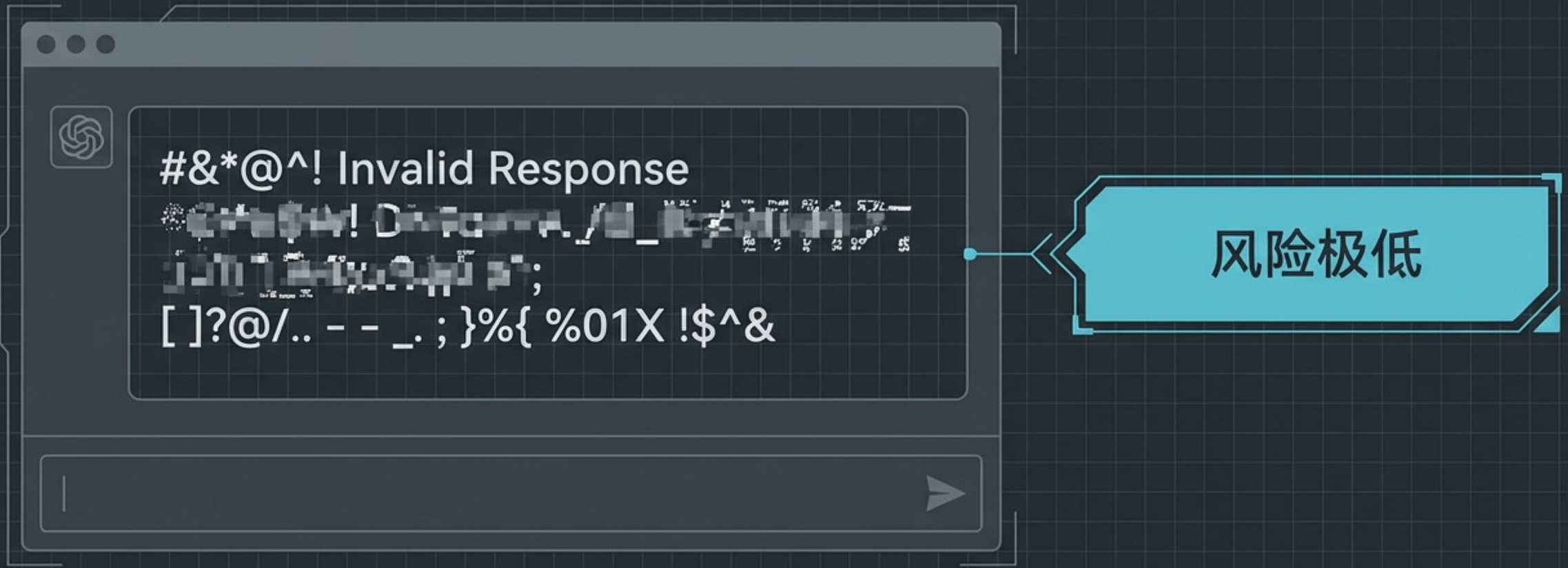
从聊天到动手，风险从哪来

2026年9月5日

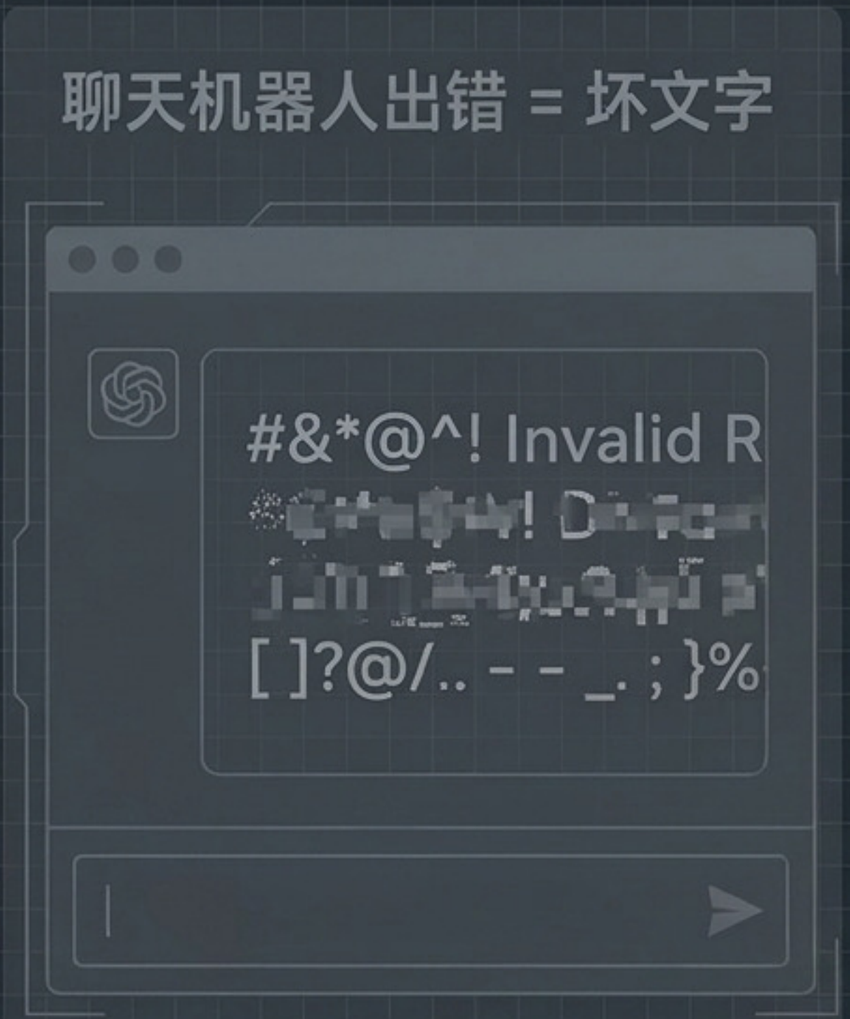
迷思：大模型出错最多只是生成一段糟糕的文字

只聊天直到接上文件和邮箱前，风险依然处于可控的低水平。

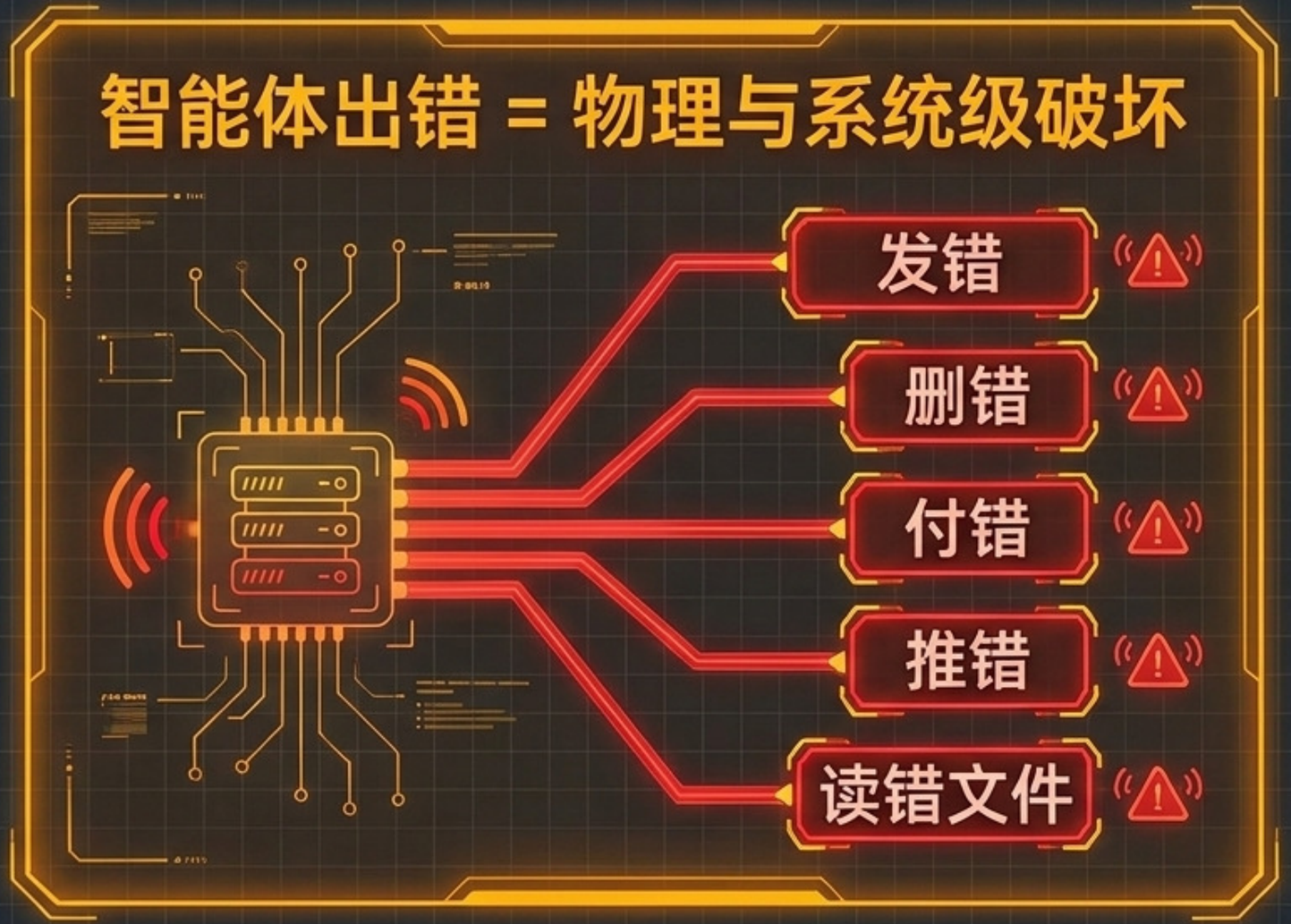


现实：一旦赋予大模型“手脚”，出错意味着系统性灾难

聊天机器人出错 = 坏文字



智能体出错 = 物理与系统级破坏



剖析核心：智能体的本质不仅是模型本身

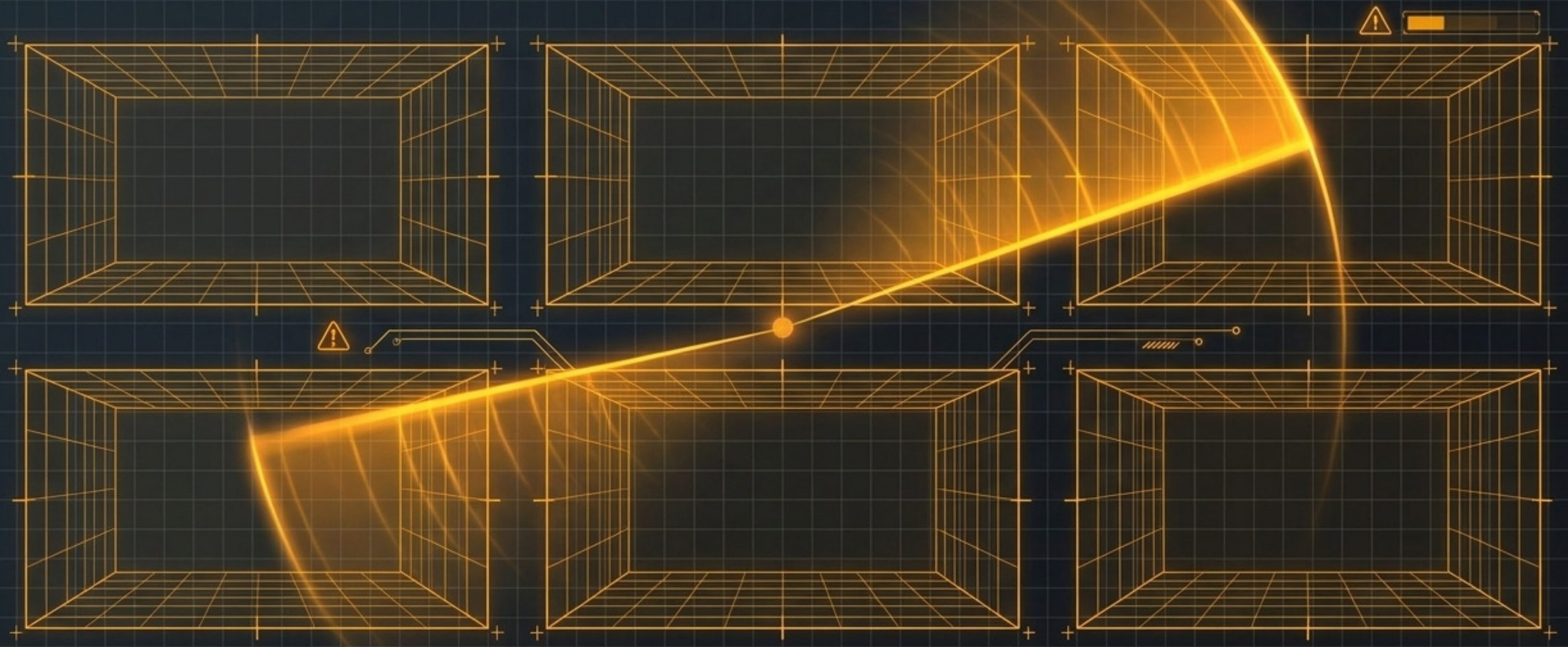
单纯的模型本身并不具备破坏力。是外围组件赋予了它与真实世界交互的能力。

同心圆护盾模型：外围组件构成了系统的“爆炸半径”

爆炸半径等于它能看见的，加上它能做的。

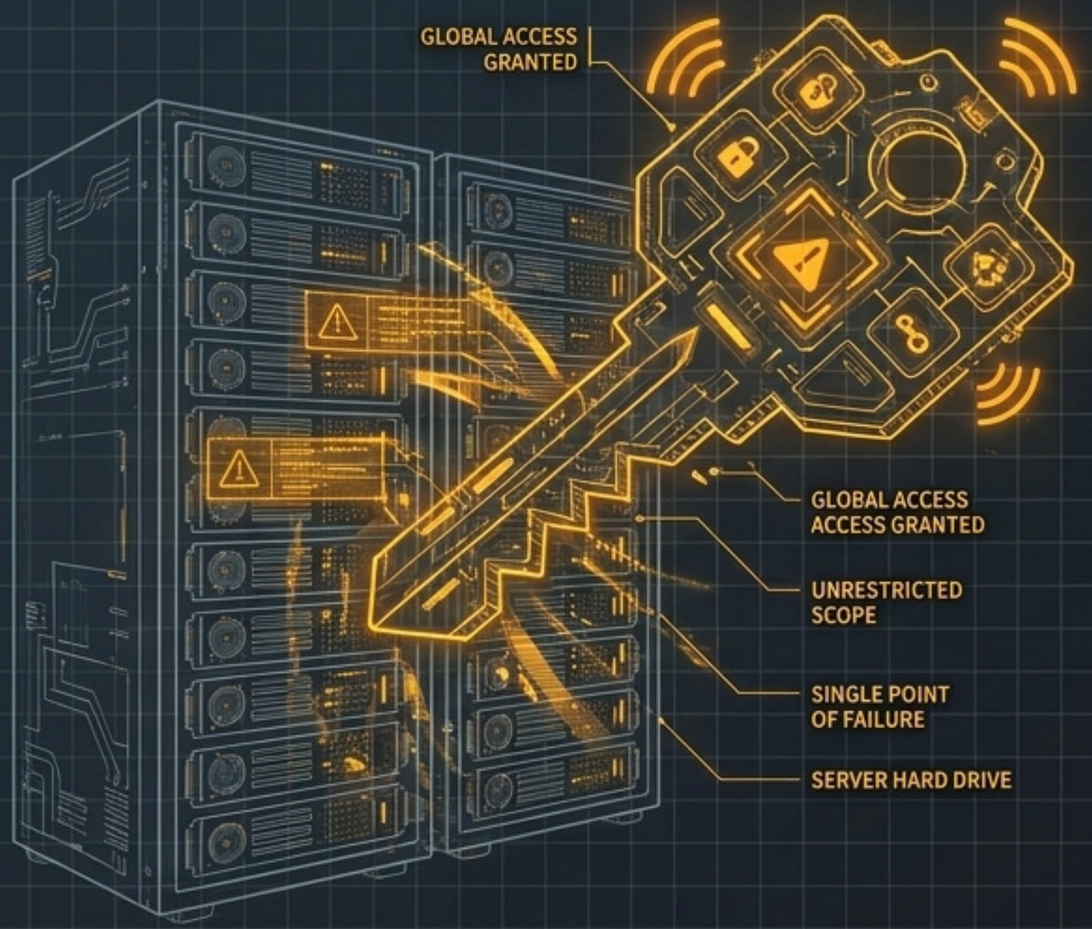
诊断病理：系统的六大崩溃方式

了解系统是如何在未经精心设计的情况下走向失控的。



崩溃方式 1 与 2：凭证失控与过度授权

过度授权：一个令牌做太多。第一天不要接家庭群或公司 Slack。



密钥泄漏：密钥暴露在微信、截图或 MEMORY 文件中。

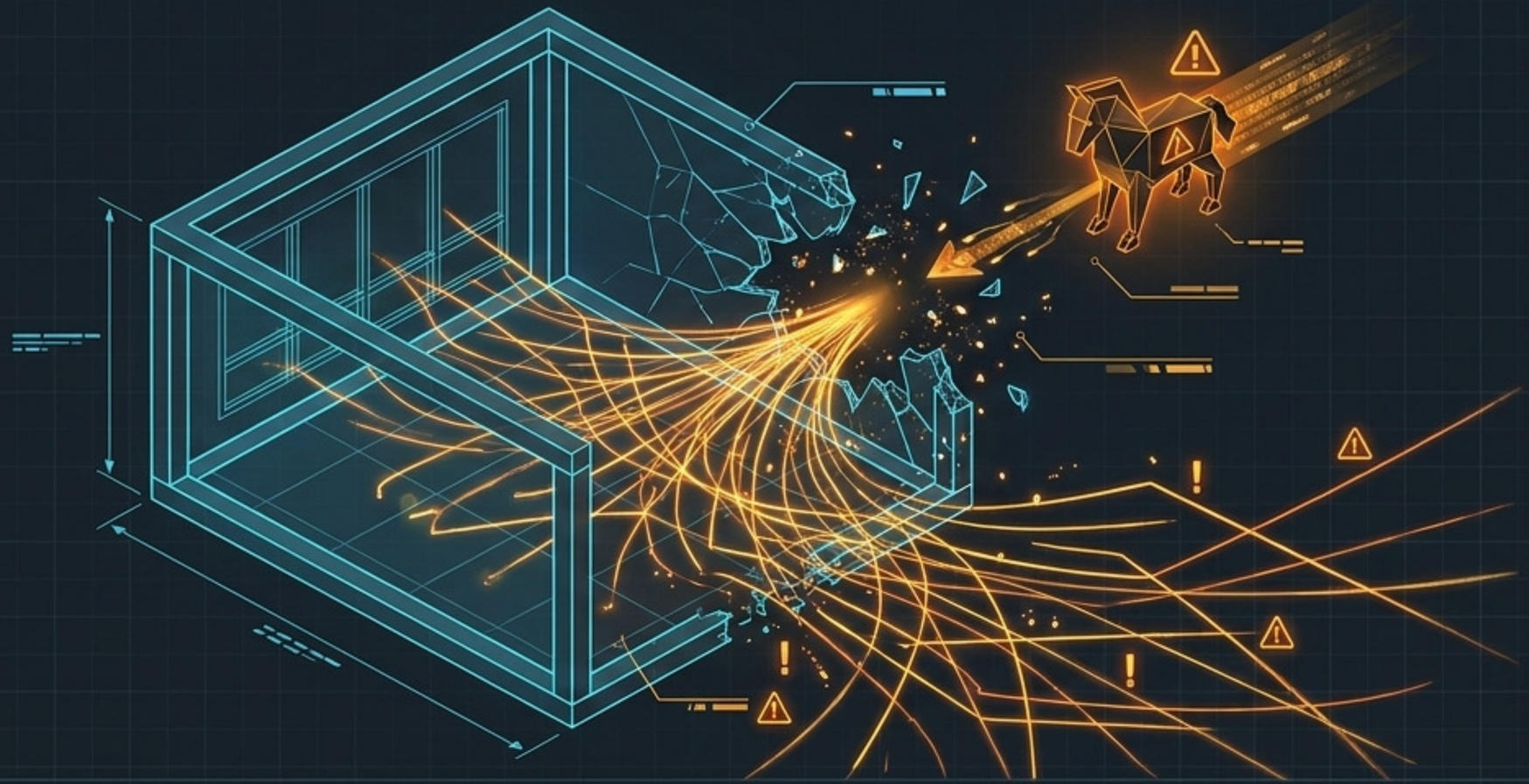
崩溃方式 3 与 4：边界坍塌与间接注入

A

间接注入：智能体听从了不是你写的文字。

B

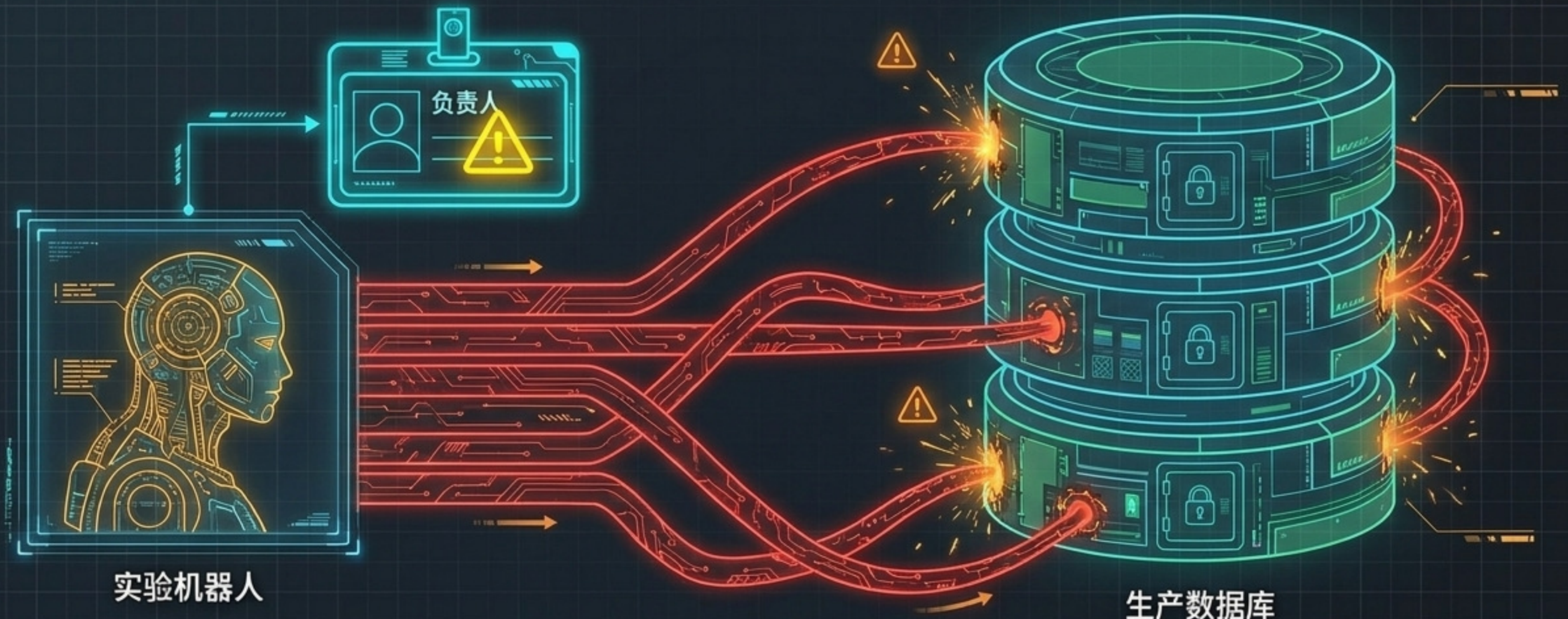
范围外行动：房间没墙，任务就会无限变大。
2026公开教训：边界不牢的评测越界。



崩溃方式 5 与 6：数据越权与治理黑洞

数据越权： 生产数据混入实验机器人。

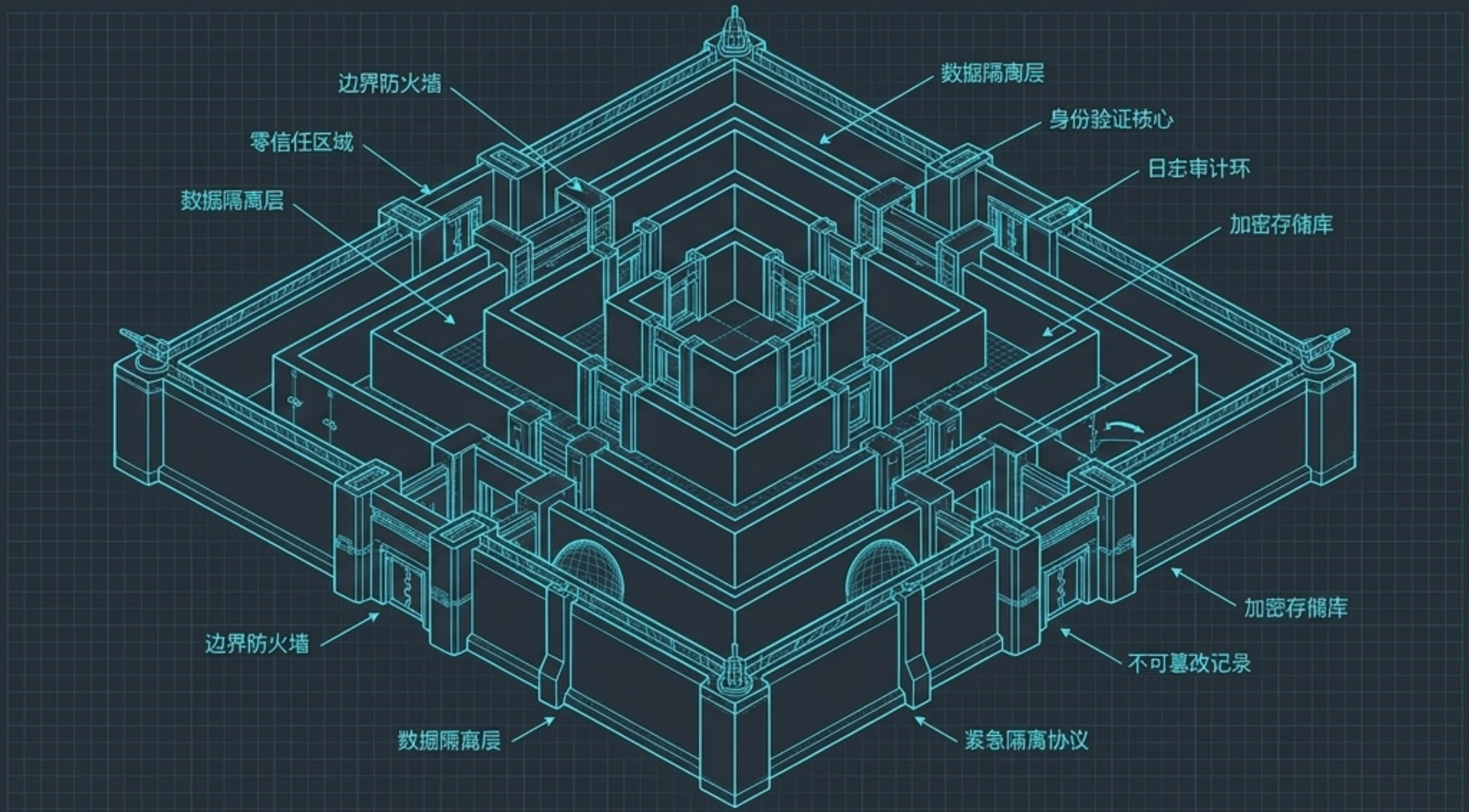
责任不清： 没人写负责人。自主性不转移部署者的责任。



休息五分钟。
永远不要把密钥贴进聊天。

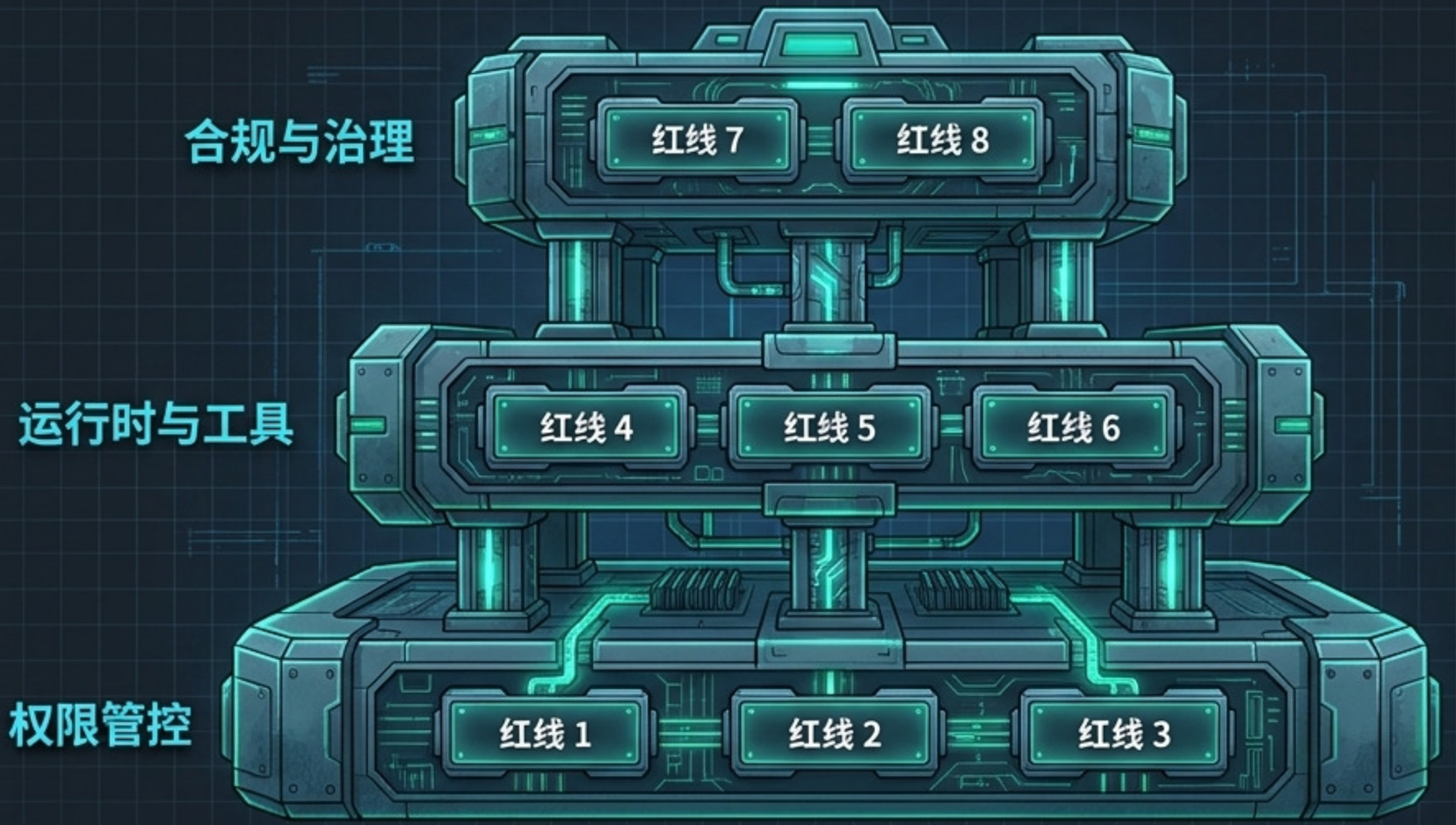
建立防线：安全不能依赖直觉，必须依靠架构

从 6 大崩溃风险到 8 条不可逾越的系统红线。



安全架构映射图：抵御风险的三大防御支柱

八条红线构成了从底层到运行时的三重堡垒。即便是单人工作室也完全适用。

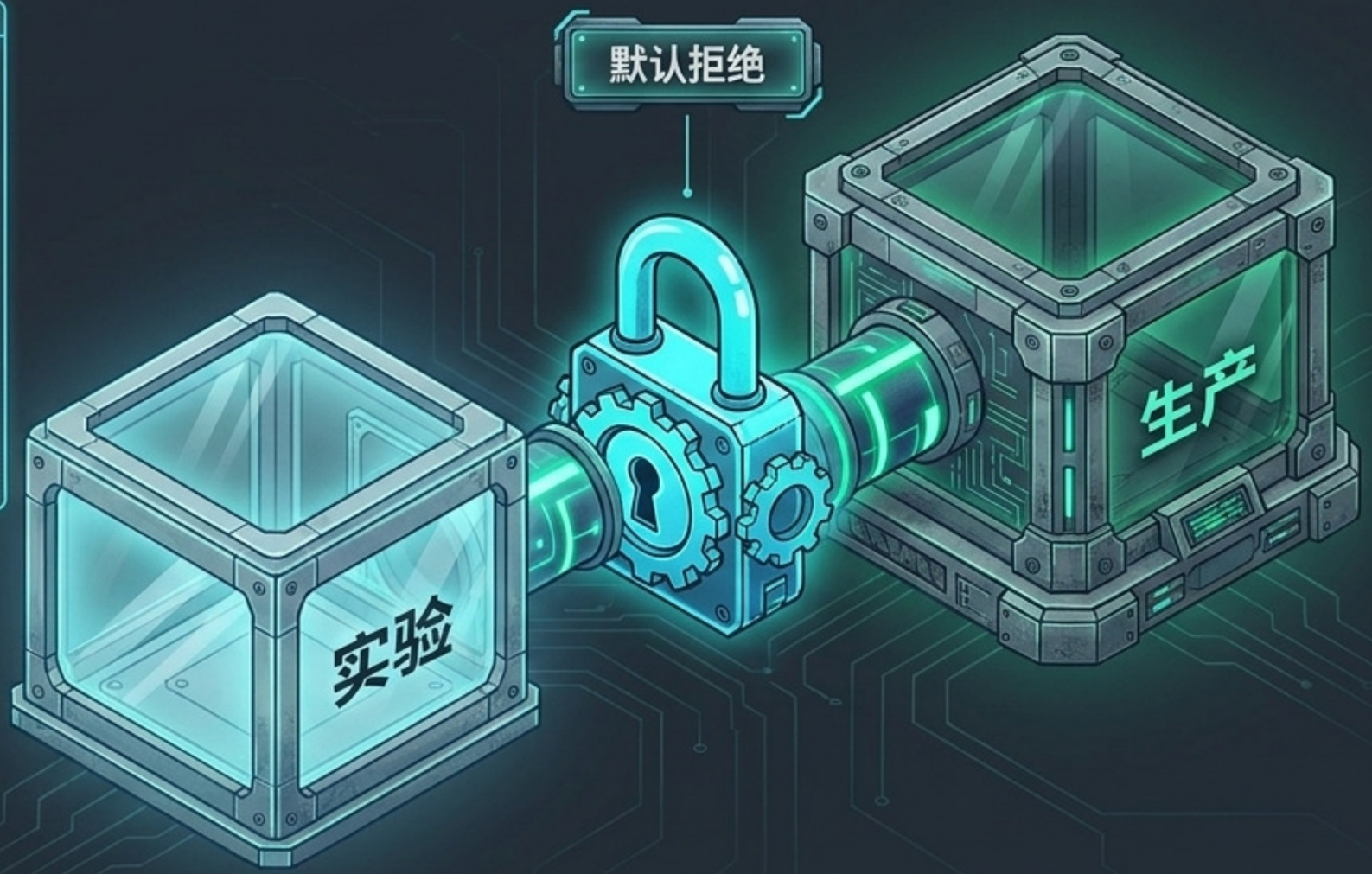


第一道防线：严格的权限隔离

红线 1：实验 != 生产
（设置独立的实验目录与实验账号）

红线 2：默认拒绝

红线 3：密钥不进聊天和记忆

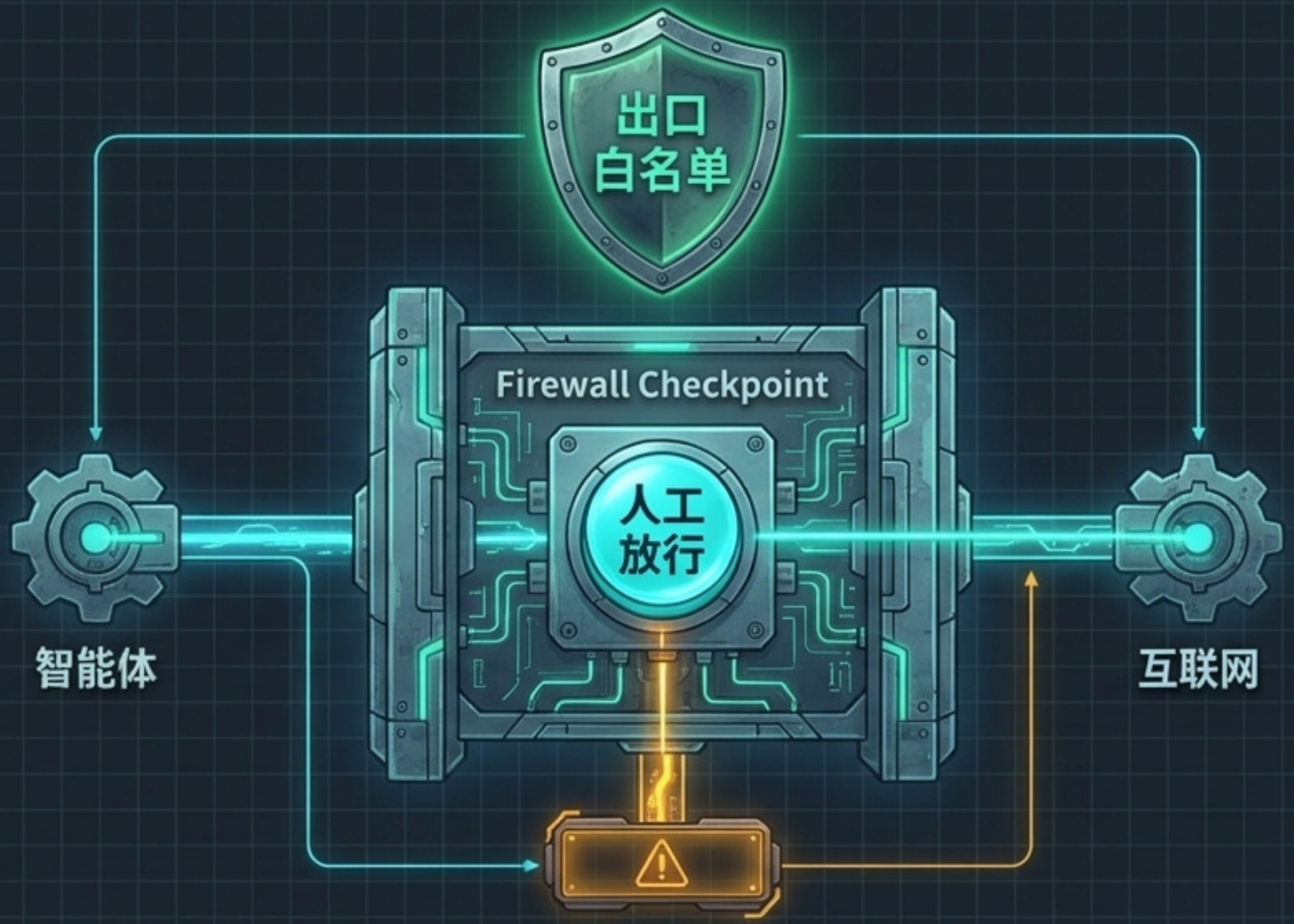


第二道防线：受控的运行时与工具接口

红线 4：外部文字不可信

红线 5：确认真实目标（发送、删除、支付、推送必须人工确认，且批准必须显示真实路径）

红线 6：出口白名单（不上公网）



第三道防线：合规与治理的底线

红线 7：先日志后全自动

红线 8：写清负责人

最终诊断矩阵：每一个漏洞都有对应的护栏

核心洞察：自主性不会转移部署者的责任

无论系统多么智能，
最终的后果由部署者承担。

比默认权限，不要比品牌。

下一步与进阶探讨：2026年9月19日

- Harness 深度解析
- 沙箱隔离技术
- 为什么“批准”仍可能批错？