移动 Agent 安全技术白皮书

目　录

序 言

当今世界，以人工智能技术为代表的新一轮科技革命与产业变革方兴未艾。随着计算能力的飞跃、算法的持续创新以及海量数据的积累，人工智能正以前所未有的深度和广度融入经济社会发展的方方面面。在这一浪潮中，移动智能体（Agent）作为智能终端上一种新兴的 AI 应用形态，凭借其高度的情境感知、智能决策与自主任务执行能力，在办公效率提升、娱乐体验革新、医疗健康服务、金融智能风控等众多领域展现出巨大的应用潜力和价值，成为推动千行百业智能化升级的重要赋能者。

然而，机遇与挑战并存。随着移动 Agent 的复杂性和自主性不断增强，其安全与合规风险也日益凸显。数据隐私泄露、算法决策偏见、恶意指令攻击、以及难以预测的自主行为等新型安全问题，给个人权益、企业运营乃至社会公共安全带来了前所未有的威胁。如何在这场技术变革中，既能充分释放创新活力，又能确保其发展过程安全、可控、可信，已成为全球产业界和学术界亟待解决的共同核心议题。

在此背景下，本白皮书旨在对移动 Agent 所面临的安全挑战与关键技术进行系统化的梳理、研究与分析。我们不仅致力于厘清移动 Agent 的技术脉络与安全现状，更希望构建一个面向未来的、多层次的安全技术框架，为产业健康发展提供清晰的建设思路与发展建议。

全书共分为四个部分：第一章将全景式地描绘移动 Agent 的应用现状与技术框架，并深入剖析其面临的核心问题；第二章将从政策法规、技术趋势、专利布局、标准演进及产业生态等多个维度，全面把脉移动 Agent 安全的发展态势；第三章将聚焦核心，提出一个分层式的安全技术框架，并阐述构建关键安全能力的思路；第四章则将基于前述研究，凝练产业面临的紧迫挑战，并最终提出具有前瞻性和可操作性的发展路径与建议。

我们期望本白皮书能够为行业开发者、企业决策者、政策制定者以及所有关注移动 Agent 发展的同仁们提供一份有价值的参考，共同推动移动 Agent 技术在创新与安全的平衡中行稳致远，为构建一个智能、高效、安全、可信的数字未来贡献力量。

在此衷心感谢参加本报告编写的各单位、组织及个人。

牵头单位： OPPO 广东移动通信有限公司

参与单位： 华中科技大学、蚂蚁金服、火山引擎、阿里云、复旦大学、西安交通大学、南京邮电大学（排名不分先后）

参与人员： 杨明慧、王浩宇、王超、吴良轩、王蒙、王宁、邱若男、刘天铭、王美珍、罗元海、陈景远、韩方、万小飞、文君、马绍青、张源、蔺琛皓、黄海平、蔡杰、刘森、杨旭、林志泳、周亚琴等（排名不分先后）

虽然我们编委的每一位专家都努力地尽量把工作做到尽善尽美，但白皮书仍不可避免有各种纰漏，恳请行业内专家和同仁不吝赐教。

摘　要

随着云端、人工智能、物联网等各类技术手段的广泛使用，移动智能终端已经成为人们日常生活的必备工具。终端内存储的重要数据和隐私信息规模不断扩大，移动智能终端行业也成为保护用户数据和隐私安全的最前线。

在万物互联的浪潮中，终端设备的连接数量和数据规模达到了前所未有的程度。尽管业界已引入相关安全防护技术，但针对终端和应用的攻击事件依然频繁发生；同时，应用软件功能日益复杂、耦合度不断增加，远远超出传统网络安全防御体系的覆盖范围。

随着全球个人信息保护和数据安全法规的不断强化，用户对隐私保护的要求在显著提高。与此同时，AI Agent 应用的需求迅速增长，其复杂性与可用性要求不断提升，使软件系统更庞大、更脆弱，运行中故障或失效的风险加大，可能对用户造成直接或间接损害。尽管业界已提出多种 Agent 安全技术与框架，但安全与隐私事件仍频繁发生，这表明移动 Agent 安全研究仍处于探索阶段，亟需系统化的方法识别痛点、总结行业现状并明确未来趋势，从而逐步构建抵御恶意用户、恶意软硬件与系统攻击的综合防护能力。

移动 **Agent** 已成为现代软件技术发展的重要趋势和必然选择，其运行过程中的**系统可靠性、身份可信性、行为可信性、数据安全性**已经成为安全研究的核心问题。通过引入**机密计算、可信身份、行为分级与权限管理、数据流转安全与透明交互**等概念和技术，可以有效分析和约束 Agent 行为，使其运行结果符合预期，成为构建可信移动 Agent 安全体系的重要手段。

本白皮书重点聚焦于移动 AI Agent 安全技术的**四个关键维度**：

1. **机密计算系统**—— 包括端侧安全计算、传输安全、云端安全计算与可信证明，确保计算和通信环境的可信与隔离；
2. **可信身份基础设施**—— 提供 Agent 唯一可信身份认证与工具绑定，保障跨域交互过程中的身份可信与可追溯；
3. **细粒度权限管理**—— 通过权限最小化原则与动态管控机制，对 Agent 行为进行有效约束与异常检测；
4. **数据流转安全与透明交互**—— 保障数据在跨终端、跨应用、跨组织中的流转安全，并为用户提供可感知、可追溯的透明交互能力。

在每个维度中，我们总结了它们在不同系统中的应用实践，并探讨了不同维度之间的协调与冲突。同时，我们还对移动终端 Agent 安全技术未来可能的发展方向进行展望。

1 移动智能体 Agent

移动 Agent 在智能终端设备中常以应用程序的形式存在，通过感知周围世界并使用可用的工具来实现其目标。随着 **LLM（Large Language Model）**在关键能力上的不断成熟，Agent 可以理解复杂输入、推理和规划、可靠地使用工具，人们越来越多的将 Agent 应用到生产环境中。在本白皮书中讨论的移动 Agent 指的是基于生成式 AI 模型实现的 Agent。

1.1 LLM 与 Agent 的关系

LLM 与 Agent 既紧密相关，又存在抽象层次上的差异。LLM 本质上是一种基础智能模型，能够提供强大的语言理解、生成与推理能力；而 Agent 则是在 LLM 之上构建的应用形态，通过循环交互、环境反馈、工具调用和状态记忆等机制，将模型的静态能力转化为动态的任务执行力。

换言之，LLM 是“能力基座”，Agent 是“应用载体”。LLM 可独立完成对话或单步任务，但当需要跨应用、跨模态的复杂操作时，必须通过 Agent 框架将其与外部工具、系统 API、环境感知机制结合，才能完成端到端任务。

1.2 移动 Agent 组件

Agent 行为、行动和决策的基础组件的组合实现了 Agent 认知架构。移动 Agent 是智能终端系统中的关键组成部分，具备自主感知、决策和执行能力，根据系统架构，典型的移动 Agent 组件可以分为三个主要层次：**编排层**、**模型层**和**感知执行层**。图 1 展示了典型的 AI Agent 组件构成，各层次的功能及作用如下所述：

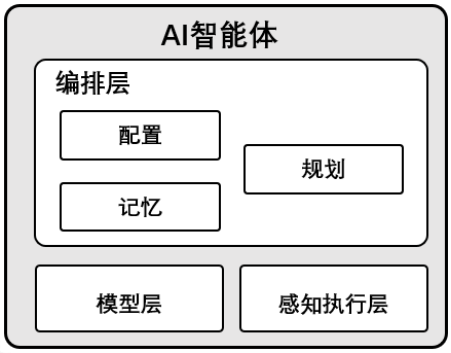

图 1 典型的 AI Agent 组件

（1） 编排层

编排层负责系统的管理与调度，主要包含：

- **配置**：初始化系统环境与设备设置。
- **记忆**：保存与回放历史数据，支持决策过程。
- **规划**：根据目标任务和资源状况，制定执行计划。

（2）模型层

模型层通过 LLM 进行推断和决策。它处理来自感知层的数据，并生成相应的行动指令，指导 Agent 进行任务执行。

（3）感知执行层

感知执行层与设备的硬件和传感器直接交互，完成对外界环境的感知和对设备的控制：

- **感知**：通过设备的传感器（如加速度计、GPS、摄像头、触摸屏等）获取用户和环境数据，帮助 Agent 理解当前状态。
- **执行**：根据感知数据与决策层的指令，控制设备执行任务，如调整屏幕亮度、触发震动或与应用交互等。

这三个层次协同工作，使得移动 Agent 能够在各种场景下智能感知、决策与执行任务，提供个性化和高效的用户体验。

1.3 移动 Agent 架构

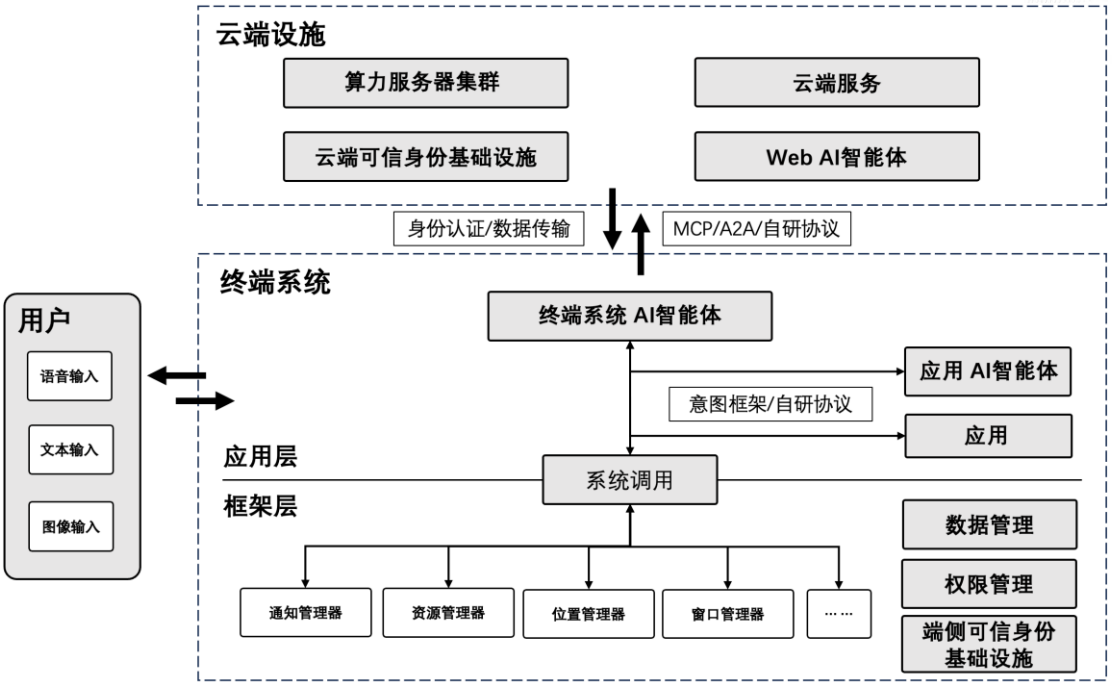

图 2 移动 Agent 架构

图 2 展示了典型的移动 Agent 架构，其中主要分为三个部分：**用户**、**云端设施**和**终端系统**。每个部分负责不同的功能，共同支持智能体的工作流程。

（1）用户

用户是与 AI Agent 交互的主体，提供不同的输入方式（如语音、文本、图像）。用户不仅可以通过这些方式输入请求，还可以查看执行过程的可见性，了解即将执行的动作和数据使用情况。用户可以根据需要确认、暂停或终止操作，并有权导出或删除个人数据与记忆，确保数据隐私与控制。

（2）云端设施

云端设施通过 MCP、A2A 等协议与终端系统进行连接和数据交换。云端服务提供强大的计算能力和 Web AI 智能体支持，处理一些较为复杂的计算任务，并将结果返回至终端系统。云端设施保证了数据的高效处理与资源的集中管理。

（3）终端系统

终端系统是移动 Agent 的实际执行环境，主要包含以下几个层次：

- **应用层**：终端上的智能体通过与云端服务和应用层应用的交互，执行特定任务。
- **框架层**：框架层负责系统调用和能力访问，确保智能体能够访问所需的系统功能，同时进行访问控制和数据管理，保障用户数据安全与隐私。

通过上述架构，移动 Agent 能够高效地完成任务，确保数据安全，并提供个性化的用户体验。

2 移动 Agent 安全技术发展态势

随着 LLM 在自然语言处理、代码生成等多个领域的广泛应用，其在移动计算环境中的集成能力也持续增强，正推动智能移动 AI 智能体成为新一代人机交互和任务自动化的关键形态。这类系统通常集成自然语言理解、多模态感知与动作执行能力，通过对用户意图的解析与上下文的建模，在多应用之间完成高度自动化的任务编排，显著提升了终端设备的智能化水平。相较于传统的语音助手，现代 AI Agent 更加主动、持续并具备状态感知能力，能够基于少量交互完成复杂指令执行，在人机交互范式上实现了质的飞跃。

在此背景下，本报告将从政策法规、专利与标准、学术与产业研究进展以及产业图谱等多个维度进行系统梳理与分析，为部署和应用移动 AI Agent 时提供安全参考与战略指引。

2.1 政策法规

随着人工智能技术的快速发展，其在经济社会各领域的应用不断深化，同时也带来了伦理、安全、隐私和治理等新挑战。全球主要国家和地区纷纷加快构建人工智能治理体系，在推动技术创新的同时，逐步建立监管框架，引导人工智能向可信、可控、可持续的方向发展。

欧盟

- 2018 年发布《人工智能创新计划》、2019 年提出《可信人工智能伦理指南》，2020 年推出《人工智能白皮书》，逐步形成治理蓝图。
- 2021 年提出《人工智能法》，成为全球首部综合性 AI 监管法。
- 2024 年进一步出台创新计划，提供资金支持初创企业。

美国

- 2023 年出台《人工智能风险管理框架》及第 14110 号行政令，从国家安全到产业应用全面部署 AI 风险管理。
- 2024 年继续推动相关指南和框架，强化生成式 AI 安全治理。
- 2025 年新政府上台后转向“轻度监管”，提出加速产业发展的政策。

日韩

- 2024 年韩国通过《人工智能基本法》，设立专门机构推动产业发展。
- 2024 年日本发布《人工智能业务指南》，并在 2025 年出台《人工智能技术研发及应用促进法案》，但监管约束力有限，更注重发展。

国际合作

- 2023 年，中国提出《全球人工智能治理倡议》，并与多国共同签署《布莱切利 AI 宣言》。
- 2024 年，欧盟通过全球首个具有法律约束力的国际公约《人工智能与人权、民主和法治框架公约》，联合国也强调全球 AI 安全与标准建设。

- 2025 年，由中国、法国、印度等 61 个国家在巴黎 AI 行动峰会上共同签署《巴黎人工智能宣言》，确立了以开放、包容、道德为核心的原则，旨在构建一个普惠、安全、可持续的全球人工智能治理框架。

中国

- 2017 年《新一代人工智能发展规划》首次将 AI 提升至国家战略。此后多部行动计划和产业政策持续推进 AI 发展。
- 2021－2024 年间，陆续出台《生成式人工智能服务管理暂行办法》《人工智能生成内容标识办法》等专项法规，构建从顶层设计到专项治理的多层次法律体系。
- 2025 年“人工智能+”行动纳入《政府工作报告》，地方政府同步推出创新政策，推动产业与安全落地。

全球人工智能治理正呈现出“战略先导—法律落地—国际协同”的时间演进特征，为移动 AI Agent 的安全研究和产业应用提供了制度边界与发展指引。

2.2 专利和标准分析

2.2.1.AI Agent 相关专利

从专利申请与授权情况来看，AI Agent 技术仍处于前沿探索阶段，整体数量有限，主要集中在应用层面。例如：

- 基于多智能体深度强化学习的多层卫星网络路由方法；
- 基于多智能体近端策略优化的雷达航迹规划方法；

这类专利反映了多智能体与大模型在垂直行业中的结合，但端侧 Agent 的应用布局与技术路线仍待拓展，高校与企业在该领域的创新空间巨大。

在安全方向，已有部分专利聚焦于大模型与 Agent 的安全防护，如：

- 基于异构多核处理器的 AI 安全防御方法；
- 基于思维链的大模型安全防护方法等。

这些专利涵盖了硬件安全、风险评估、模型安全机制等多方面，为未来 Agent 安全技术的发展提供了有益探索。

2.2.2 AI Agent 相关标准

从标准体系来看，国内外 AI 相关标准主要集中在通用技术与应用层面，涉及安全的部分相对有限。现有标准多强调安全要求、伦理设计、安全测试与评估，但在可实施性与细化程度上仍有不足。

国际标准化组织（ISO）与国际电工委员会（IEC）已陆续启动 AI 安全相关标准的制定工作，其中部分标准与 AI Agent 的安全治理相关（如表 1 所示）。总体而言，标准化进程仍处于早期阶段，需要进一步细化针对 Agent 的安全规范、测试方法和合规框架。

表 1　相关国际标准

分类	标准号	标准名称
数据管理	ISO/IEC 5259-3:2024	人工智能-分析和机器学习的数据质量.第 3 部分：数据质量管理要求和指南
风险管理	ISO/IEC 42005:2025	信息技术-人工智能-AI 系统影响评估
	ISO/IEC 23894:2023	信息技术-人工智能.风险管理指南
AI 管理	ISO/IEC 23053:2022	使用机器学习的人工智能系统框架
	ISO/IEC 42001:2023	信息技术-人工智能-管理系统
	ISO/IEC 38507:2022	信息技术治理-组织使用人工智能的治理影响
隐私安全	ISO/IEC 27091	人工智能-隐私保护
	ISO/IEC 27090	人工智能-解决人工智能系统中安全威胁和故障的指南
内容与功能安全	ISO/IEC TR 5469:2024	人工智能-功能安全和人工智能系统
	ISO/IEC TR 24028:2020	人工智能-人工智能中可信赖性概述
	ISO/IEC TR 24368:2022	信息技术-人工智能-伦理和社会问题概述

国内 TC260、CCSA 等 组织也在不断推动 AI 安全标准的建立，现行或即将实施的可能对于 AI Agent 领 域有参考意义的国家标准如表 2 所示。

表 2　相关国家标准

分类	标准号	标准名称
大模型管理	GB/T 45288.1-2025	人工智能 大模型 第 1 部分：通用要求
	GB/T 45288.2-2025	人工智能 大模型 第 2 部分：评测指标与方法
	GB/T 45288.3-2025	人工智能 大模型 第 3 部分：服务能力成熟度评估
AI 管理	GB/T 45081-2024	人工智能 管理体系
生成式 AI 安全	GB/T 45674-2025	网络安全技术 生成式人工智能数据标注安全规范
	GB/T 45654-2025	网络安全技术 生成式人工智能服务安全基本要求
	GB/T 45652-2025	网络安全技术 生成式人工智能预训练和优化训练数据安全规范

2.3 技术进展

2.3.1 学术研究进展

在学术界，已有研究综述了大语言模型（LLM）在需求工程、代码生成、软件测试等方面的潜力与局限，指出其在移动应用 GUI 测试中表现突出，尤其在测试输入生成任务中展现出较强的语义理解与推理能力，为移动 Agent 研究提供了理论支撑[2]。

在 **GUI 自动化测试**方向，LLM 被用于自动生成语义相关的输入，显著提升测试效率和缺陷发现率（如 QTypist[3]、GPTDroid[4]、AutoDroid 系列[5][6]）。这些方法通过上下

文建模、功能感知推理及代码生成机制，实现了更高效的测试与任务自动化。

在**多模态理解与动作建模**方面，大型动作模型[7][8] 将视觉输入与指令结合，模拟类人 UI 操作，在界面识别和行为规划上取得进展，并在一定程度上缓解了幻觉问题[9][10]。

在 **Agent 安全研究**方面，移动 Agent 的攻击面逐渐显现。杨玉龙等人[11] 总结了 34 种新型攻击，涵盖机密性、完整性与可用性三大类，并通过多模态载荷展示跨模块渗透的可能性。吴良轩等人[12] 则提出了一套半自动化测试工具，对 9 个主流 Agent 从大语言模型交互、系统交互与 GUI 交互三个维度开展了 11 类攻击实验。

业界普遍将智能体安全挑战归纳为五类：

a. **指令层攻击**：恶意输入误导智能体，如越狱指令、提示词注入、对抗样本；已有 AgentScan[12] 对此类攻击的可行性进行验证。

b. **界面元素扰动**：通过覆盖、混淆等方式篡改界面，陈超然等人[13] 系统研究了 GUI 恶意注入问题。

c. **权限与工具误用**：Agent 滥用终端权限或 API 接口（如位置权限、网络请求），存在隐私泄露风险[13]。

d. **供应链攻击**：在模型、库、SDK 的开发与分发环节植入后门，导致 Agent 行为失控。

e. **通信与协议层攻击**：对 MCP、A2A 等协议的交互进行劫持，常见攻击包括工具投毒、傀儡攻击、Rug Pull 攻击等。

综上所述，学术界正积极探索 LLM Agent 在移动端的能力边界与部署方式，同时也意识到安全风险的复杂性。然而，目前仍缺乏统一的能力分类框架、权限调度模型与行为约束机制，这为本项目的研究提供了紧迫的价值与现实意义。

2.3.2 工业界技术进展

在产业界，移动 AI 智能体已成为智能终端演进的重要方向，并在多个生态系统中快速落地。根据部署方式和集成深度，当前主要有三类实现路径：

- **系统级智能体**：由设备厂商或操作系统平台开发并深度集成，具备高执行权限和低延迟响应能力，代表产品包括 OPPO 小布助手、 Honor YOYO 助手、Vivo 蓝心小 V、与 Apple Siri 等。
- **第三方通用智能体**：如智谱 AI 的 AutoGLM，作为普通应用运行在标准权限模型下，具备良好的跨平台兼容性。
- **智能体框架型解决方案**：如阿里巴巴 MobileAgent[16][17]、腾讯 AppAgent[18][19]，采用客户端—服务器架构，通过 ADB 等接口实现全栈控制。

在**协议**层面，国内外企业正加速布局。阿里云“百炼”上线业界首个全生命周期 MCP 服务，腾讯云发布支持 MCP 插件托管的开发套件，Cursor、OpenAI、Winsurf 等海外厂商也宣布接入 MCP 协议。通信协议逐步趋同，但在**认证协议**上仍高度碎片化：一类是基于

OAuth 2.0 / OpenID Connect 的企业级认证，安全性高但延迟较大、跨平台互认不足；另一类是轻量化 API Key 认证，接入便捷但静态密钥管理存在漏洞，难以支撑高价值场景。

2.4 产业图谱

2.4.1 AI Agent 生态

随着生成式 AI 技术在移动端的快速渗透，移动 AI Agent 已成为重塑移动应用交互与生产力的重要入口。根据开发主体和系统权限，当前产品生态主要可分为三类：

1. **新兴 Agent 框架**：由高校或研究组织开发，采用客户端—服务器架构，通过 ADB 等工具实现全栈控制，强调可行性研究与多步骤、多模态任务处理，代表如阿里巴巴 MobileAgent、腾讯 AppAgent。
2. **系统级 AI Agent**：由设备厂商或操作系统平台开发，深度集成在设备原生系统中，具备高权限访问系统 API 和内部接口，代表如 OPPO 小布助手、荣耀 YOYO 助手、Vivo 蓝心小 V、Apple Siri 等。
3. **第三方通用 Agent**：由独立软件厂商开发，以标准 Android 应用形式提供通用 AI 服务，如 Manus、智谱 AutoGLM，具备跨平台兼容性。

2.4.2 Agent 协议生态

MCP（Model Context Protocol）作为工具调用核心协议，为 Agent 提供跨平台、安全可控的工具集成能力。2025 年起，国内外模型厂商和云服务商纷纷接入 MCP：

- **开发侧**：阿里云百炼推出全生命周期 MCP 服务；腾讯云发布支持 MCP 插件的开发套件与大模型知识引擎；华为云推出原生应用运行平台 + MCP 方案；微软、OpenAI、Google DeepMind 也宣布深度集成。
- **应用侧**：MCP Server 市场快速扩张，覆盖支付、地图等高阶能力。如百度地图接入 AI 全局感知 MCP 协议，支付宝推出支付 MCP Server，蚂蚁集团设立 MCP 专区，360 翎林 AI 发布 MCP 万能工具箱，接入超 110 款工具。

这种“协议—平台—场景”三级协作模式，使 Agent 能够无缝复用支付核身、地图 API 等能力，实现从工具孤岛向产业级智能协作网络的升级。安全方面，MCP 衍生出补充协议 MCIP，由香港科技大学与华为提出，通过结构化日志和安全守护模型提升风险检测与防御能力。

此外，业界在通用标准协议方面百家争鸣，各类协议及其特点可参见表 3。

- Google 发布 A2A 协议，支持智能体间通信与事务协作，已开源；
- AComP 支持异构智能体技术栈互操作；AConP 定义 Agent 调用与配置接口；元协议 Agora 允许 Agent 在不同上下文采用不同通信协议；
- 国内社区推出 ANP 协议，基于 P2P 架构与 W3C DID 标准实现去中心化身份管理，并支持元协议层智能协作。

表 3 Agent 协议概览

协议名称	开发者	应用场景	认证方式
Model Context Protocol	Anthropic 公司	Agent 与工具和数据源的交互	OAuth
Agent-to-Agent Protocol	谷歌公司	跨 Agent 间协作	OAuth
Agent Network Protocol	ANP 社区	分布式 Agent 网络通信	W3C DID
Agent Communication Protocol	IBM 公司	企业内 Agent 系统集成	OpenAPI
Agent Connect Protocol	Langchain 、Galileo、Cisco 等公司	Agent 调用标准接口	OpenAPI
Agora	牛津大学、Eigent AI 等	Agent 之间的元协议	-
Language Model Operating System	Eclipse 基金会	物联网与 Agent 的网络通信	-

3 移动 Agent 安全技术框架和生态

3.1 Agent 安全技术架构

图 3 智能体安全技术框架

智能体（Agent）安全技术框架，自下而上分层构建安全体系，各层作用如下：

- **设备和基础设施层：**作为底层支撑，依托安全硬件（芯片、TEE 等）、机密计算系统（端侧与云侧）和密码学基础设施（KMS/HSM、PKI 等），为智能体运行提供可信的物理与计算环境。通过设备/环境证明、密钥安全存储与加密传输，降低底层被篡改与数据泄露风险。
- **系统和平台层：**聚焦智能体的核心安全管理。以可信身份基础设施建立身份信任根，配合行为分级与权限管理划定操作边界；通过跨域数据流转安全机制保障数据交互；通过交互透明与可控机制实现“可见、可管、可追责”。
- **协议层：**为智能体之间的交互与协作制定规则。通过意图框架明确交互语义与能力边界，采用 MCP、A2A、ANP 等协议以及 OAuth 2.0/OIDC 等授权与身份联邦机制，支撑通信、授权与调用过程的合规与可审计。
- **应用和交互层：**面向具体应用场景落地安全能力。围绕智能体身份（认证、校验）、交互（透明交互、安全通信如 mTLS）与隐私（隐私增强技术，如去标识化/差分隐私按需采用）解决用户接触面的风险；通过关键操作确认、范围与频率限制、异常反馈处理，保障交互过程稳健可控，促进智能体应用合规落地。

总体说明：各层协同覆盖从硬件到应用的全流程，系统性应对身份、通信、数据与行为等安全挑战，确保智能体在多场景中稳定、可信、可控地运行。

3.2 机密计算系统

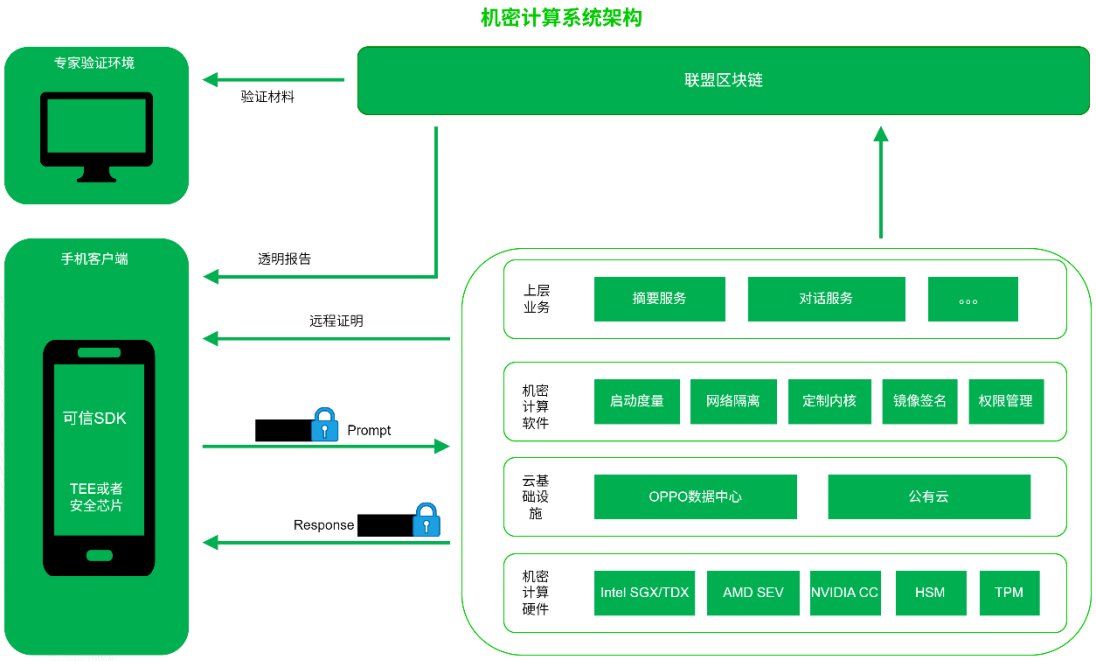

图 4 机密计算系统

移动 Agent 作为具备自主跨节点交互、动态任务执行能力的智能模块，其业务场景中频繁的端侧数据采集、跨网络传输及云侧 AI 推理，机密计算系统通过端侧安全能力、传输安全能力、云侧安全能力构建起对用户数据的全生命周期保护的长城，极大的减少了用户数据泄露、指令篡改等的可能，即使是我们 OPPO 自己也无法在用户未授权的情况下获取用户的隐私数据。机密计算系统的“端 - 传 - 云”三层安全架构，构建了全流程防护屏障。

3.2.1 端侧安全能力

端侧通过可信执行环境（TEE）为核心构建了多层安全防护体系，结合硬件隔离、加密技术、权限控制等手段，全面保障用户数据的机密性和完整性。

- **硬件级隔离：**TEE 在处理器内部创建与主操作系统（Rich Execution Environment, REE）物理隔离的“安全世界”，通过硬件机制（如 ARM TrustZone）划分专属内存、缓存和计算资源。REE 中的恶意软件或遭入侵的操作系统无法访问 TEE 内的代码或数据，即使主系统被攻破，敏感数据仍受保护。
- **数据加密与存储保护：**敏感数据仅在 TEE 内解密处理，外部传输时保持加密状态。例如 AI 用户隐私数据的加解密密钥在 TEE 内完成解密与签名，全程规避中间人攻击。TEE 将密钥与设备硬件绑定，生成设备唯一加密密钥。存储数据时，TEE 自动加密并绑定设备状态，仅当设备完整性验证通过时才能解密。根密钥永久存储于 TEE 安全区域，应用层密钥由其派生。密钥使用后立即销毁，不留痕于 REE 内存。

- **代码与执行完整性校验：**所有在 TEE 内运行的可信应用需经数字签名验证，确保代码未被篡改。启动前校验签名哈希，非法代码无法加载。从硬件固件到 TEE 系统层，逐级验证启动代码的完整性和签名，防止启动阶段注入恶意代码。
- **精细化访问控制：**TEE 内每个可信应用仅能访问其授权资源。例如生物识别可信应用无法读取支付可信应用的密钥，实现应用间隔离。敏感操作需用户主动授权（如指纹、面容识别），授权过程在 TEE 内完成，避免界面劫持。

端侧通过一系列的安全保护能力，有效避免了来自端侧恶意应用和程序的攻击，极大程度减少了终端用户敏感数据泄露的可能，给用户营造了一个安全的终端环境。

3.2.2 传输安全能力

为实现机密数据从终端到云端的端到端安全传输，仅依赖 HTTPS（TLS）是不够的。HTTPS 提供了传输层的安全，主要保护数据在传输过程中不被窃听和篡改，但它存在以下局限性：

- HTTPS 的安全性依赖于客户端库、操作系统、服务器 TLS 配置、证书颁发机构等的安全性。任何一方的妥协都可能影响整个通道。
- 不防内部威胁：服务器管理员或拥有服务器访问权限的人可以看到解密后的数据。

因此，对于高级别的机密数据，必须在 HTTPS 之上实施应用层加密，并通过远程证明方式来验证服务端的环境安全状态，保证用户数据流向的是安全的环境，以实现真正的端到端机密性。

- **目标：**确保只有发送方和预期的接收方（服务端特定组件）能够解密数据。即使传输通道（如 TLS）被攻破、服务器操作系统被入侵（非目标服务）、或中间节点（如 CDN、代理）被控制，数据本身仍是加密的。
- **原理：**终端应用在将数据放入 HTTPS 请求体之前，使用只有服务端能解密的密钥对应用层数据进行加密。服务端收到 HTTPS 请求后，需要先解密 TLS，然后由特定的、受信任的服务端组件（机密计算硬件）直接提取加密的应用层数据并进行解密。解密操作应在服务端隔离的安全环境中进行（如可信执行环境 TEE、硬件安全模块 HSM）。

HTTPS+应用层数据加密的双重加密方式，即使攻击者完全控制了网络链路或部分服务器资源，也无法获取应用层传输的机密数据明文，从而在终端到机密服务之间建立了真正意义上的端到端数据安全通道。

3.2.3 云侧安全能力

机密计算系统在云侧的安全能力是保障数据全生命周期安全的核心环节，需从硬件、平台、操作系统及协同机制等多维度构建防御体系。

3.2.3.1 机密计算硬件

机密计算硬件是云侧机密计算能力的基础，机密计算系统离不开底层硬件的支持，各家硬件计算厂商都陆续提供具备机密计算能力的硬件，常见的机密计算硬件如下：

- **Intel TDX（Trust Domain Extensions）**：通过创建隔离的信任域（Trust Domain），实现虚拟机级别的内存加密和访问控制，确保租户工作负载与云平台及相邻租户的物理隔离。
- **AMD SEV（Secure Encrypted Virtualization）**：为每个虚拟机使用一个密钥来隔离客户机和虚拟机管理程序，这些密钥由 AMD 安全处理器管理客户机更改允许虚拟机指示哪些内存页面需要加密。
- **Nvidia CC（NvidiaConfidential Compute）**： NVIDIA 机密计算可保护部署在 Hopper 和 Blackwell GPU 上的 AI 模型和算法的机密性和完整性。无论平台或工作负载在何处运行，都能通过 Blackwell 和 Hopper GPU 保持合规性并确保应用和数据在可信执行环境 (TEE) 中受到保护。
- **专用加速芯片**：如 AWS Nitro Enclaves 或 Google Titan，通过专用安全芯片实现密钥管理与硬件级可信根。

考虑到现在机密计算硬件种类繁多，单一的机密计算硬件难以短时间内难以成为行业的主流，未来很长时间内都会是多种机密计算硬件架构共存的情况，基于此现状，OPPO 机密计算系统并不绑定某一种确定的机密计算硬件架构，而是采用包容的态度去接受各种机密计算硬件，并通过硬件屏蔽层去屏蔽底层差异，为上层业务提供统一的机密服务。

3.2.3.2 云基础设施安全

在 AI 浪潮的冲击下，用户的隐私数据安全变得越来越严峻，云厂商作为 AI 云基础设施提供一个让客户放心的云上安全环境显得尤为重要，各家公有云厂商也都提供了基于机密计算硬件的云上安全环境解决方案。

- **阿里云**：基于硬件级可信执行环境（如 Intel SGX/TDX）和自研全密态技术，提供数据全生命周期加密保护，支持高性能 GPU 协同计算、动态密钥管理及远程证明，确保数据在处理过程中可用不可见，满足金融、AI、医疗等行业的严格合规要求。
- **火山云**：通过自研 HyperEnclave 技术实现兼容 RISC-V 架构的 TEE，适用于异构算力场景。
- **亚马逊云科技**：依托 Nitro System 和 AWS Key Management Service（KMS），提供端到端加密的 Enclave 工作流。
- **Google Cloud**：提供 Confidential VMs 和开源的 Asylo 机密计算框架，支持包含 Intel TDX\ AMD SEV 等多种机密计算和硬件远程证明能力，并覆盖全球几乎所有的核心区域 。

公有云在机密计算系统是尤为重要的一环，单一的企业往往难以具备全球云设施覆盖的能力，OPPO 借助各位公有云合作伙伴（阿里云、火山云、AWS、GCP 等）强大的云能力，实现全球机密计算能力的覆盖，帮助 OPPO 用户在提供 AI 智能体验的同时而不用担心用户的隐私数据遭到泄露。

3.2.3.3 运行环境安全

运行环境安全是机密计算系统安全非常重要的一部分，为了进一步防止内部人员作恶的可能，我们针对启动过程、系统内核、系统镜像、网络访问和权限管理等多个层面对操作系统运行环境进行综合安全防护。

- 启动过程：利用硬件提供的启动度量能力，从启动开始对所有运行程序进行度量，构建启动信任链，保证系统启动过程的安全。
- 系统内核：系统内核作为操作系统的核心，负责管理硬件资源、进程、内存以及提供基础的安全机制，其安全性至关重要。
- 系统镜像：机密计算系统的镜像需要进行安全定制和裁剪，只保留提供最基本业务运行所需要的软件，减少因软件程序漏洞导致的风险，所有运行的业务镜像都需要被安全签名并被验证，确保运行环境未被篡改。
- 网络访问：机密计算系统中的网络被严格控制，所有非业务网络请求会被记录和禁止，任何人无法访问和探测到机密计算系统的环境，无论是从内到外还是从外到内的访问。
- 权限管理：业务的生产运行环境不再允许任何的人员访问（包括 OPPO 的运维和研发人员），即使是业务环境出现异常情况，我们只会保留最小必要的非用户敏感数据日志供业务开发和运维排查，业务人员仍然无法进入生产的运行环境。

通过上述多层面的安全保护，我们构建了一套有效的保护机密计算系统保护方案，极大程度降低了机密计算系统业务数据被泄露的风险，为我们的用户数据构建出一座安全的长城。

3.2.3.4 云端可信证明

可信证明机制是机密计算系统的核心技术之一，旨在通过硬件级验证来确保机密计算环境的可信性，其核心目标是通过技术手段来验证云上环境的安全性。

不同的机密计算硬件提供的证明方式也不尽相同。静态证明如 TPM/TCM 提供了一套基于平台配置寄存器度量启动阶段的组件哈希。动态证明如 TDX/SEV 的 Guest 证明，通过硬件根密钥可以实时生产加密的度量报告以供外部验证。

OPPO 结合多种机密硬件，提供异构 TEE（TPM/TDX/SEV/NVIDIA CC）的统一验证方式，解决不同机密硬件下的可信证明问题，并借助区块链和零知识证明等技术，实现对证明报告的可信验证。

通过与机密计算系统的深度融合，移动 Agent 实现了从数据采集、传输到计算推理的全生命周期安全防护，不仅为各领域的智能应用落地扫清安全障碍，更让“厂商不可见、用

户可掌控”的隐私保护理念成为现实。

3.3 Agent 可信身份基础设施

随着 Agent 功能日趋丰富，Agent 所接触的敏感数据范围和 Agent 敏感行为能力边界也在不断扩大，天然带来对 Agent 身份进行校验的安全需求。尤其在 Agent 与 Agent 之间进行协作、Agent 调用工具或功能模块等交互场景中，敏感数据和行为能力实质完成了跨域扩展，假如此类交互场景下未执行 Agent 身份校验、或身份校验强度不足，则 Agent 身份仿冒、敏感数据泄露等攻击行为的风险将急剧提升。

Agent 交互场景下，当前业界主流使用的协议主要包括 MCP、A2A、ANP 等 Agent 交互协议。这些协议在推动技术标准化中作用显著，规范了 Agent 间的交互接口与数据格式，但更多侧重接口与数据格式标准化，在 Agent 身份认证和授权方面的设计存在一些不足之处。

以 MCP、A2A、ANP 协议为例，三者在 Agent 身份认证和授权方面的分析主要如下：

- MCP
  - 提供了一个基于 HTTP 传输的身份认证和授权框架。身份认证主要使用 OAuth2.0，同时提供了一些参考的安全实践建议。
- A2A
  - 数据交互不携带身份信息，只携带凭据。复用 OpenAPI 支持的认证方式，例如 HTTP 认证（如 Basic、Bearer）、API Key、Cookie 认证、OAuth 2.0 以及 OpenID Connect 等。
- ANP
  - 数据交互携带身份信息、凭据。基于 W3C DID 标准，实现去中心化身份认证和端到端加密通信。

由上可知，MCP、A2A、ANP 三者协议在身份认证和授权方面的举措，主要是复用业界现有的身份认证协议，如 OAuth2.0 等，但是对 Agent 的具体应用场景，例如 Agent 在移动智能终端设备上应用等情况，缺少针对性的身份认证过程设计。这会增加协议适配时实施身份认证的复杂度，降低对身份冒用、数据篡改等风险的防范效果。

除了缺少针对性的身份认证设计之外，当 MCP、A2A、ANP 协议应用于终端业务场景下时，实现身份认证将存在一定的困难，由此带来更多潜在的攻击面。原因在于 MCP、A2A、ANP 三者所复用的身份认证协议主要适用于 Web 场景，而终端资源受限，且部分业务场景还存在离线协作、高频低延迟交互的特点，都对身份认证的实施过程提出了挑战。

此外，授权机制的细化程度也有待提升。MCP、A2A、ANP 协议主要描述了授权的原则和建议，但未明确规范权限申请、授予、回收等具体流程。这可能导致 Agent 越权调用敏感资源，同时增加合法 Agent 的权限管控难度。关于权限管理部分，将在 3.4 节对 agent 行为管控进行具体描述。

因此，从移动智能终端安全实践来看，MCP、A2A、ANP 协议在身份认证设计、身份认证协议在终端场景下的适配、及授权机制三个维度仍有一定的完善空间。这些协议短板将直接影响移动智能终端场景下的 Agent 业务的安全程度，形成潜在的安全风险，例如：

- 在终端内 Agent 协作场景下，身份认证适配不足可能引发身份仿冒风险、恶意 Agent 参与数据传输可能造成敏感数据篡改或数据泄露；
- 当 Agent 调用终端工具与资源时，因缺乏针对性的身份认证方案，理论上存在非法 Agent 伪装系统组件的可能，进而威胁用户数据安全与设备稳定运行。

综上，为了加强移动智能终端上的 Agent 业务安全，需要从 Agent 身份校验和数据保护的思路出发，建设 Agent 可信身份基础设施。结合手机终端业务需求，可从三方面进行考虑，如图 5 所示。

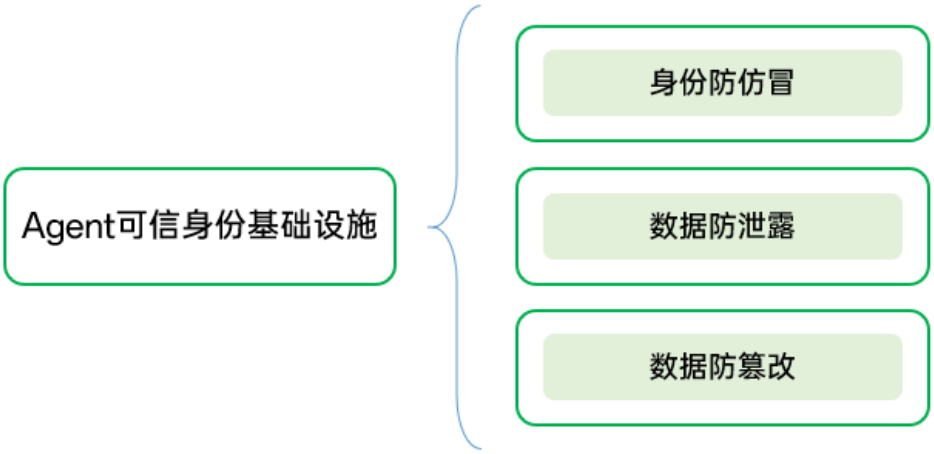

图 5 Agent 可信身份基础设施建设角度

如图 6 所示，在 Agent 可信身份基础设施的身份防仿冒环节，核心是依托合规有效的数字证书对 Agent 可信身份进行全流程的注册与校验：注册阶段需将 Agent 的唯一身份标识、属性信息与数字证书进行绑定，确保身份信息的真实性与合规性；校验阶段则通过验证证书的有效期、颁发机构合法性、证书签名完整性等信息，核对 Agent 当前提交的身份信息与注册时绑定的身份信息是否一致，最终确认 Agent 身份的可信性，为后续的数据交互环节筑牢身份信任基础。

而针对数据防泄露与数据防篡改需求，其实施前提是 Agent 已通过前述的身份认证流程 —— 在确认 Agent 身份可信后，将采用安全强度符合行业规范与业务需求的密码算法套件（如包含对称加密算法、非对称加密算法、哈希算法的完整套件），对 Agent 与 Agent 之间、Agent 与工具或功能模块之间的全链路传输数据进行双重安全处理：一方面通过数据加密操作，将敏感传输数据转化为不可直接解读的密文，防止数据在传输过程中被未授权主体窃取、窥探，保障数据的机密性；另一方面通过数据签名操作，利用 Agent 的专属密钥

对传输数据的哈希值进行签名，接收方通过验签可确认数据在传输过程中是否被篡改、替换或破坏，从而保障敏感数据的完整性，确保数据从发送端到接收端的一致性与可靠性。

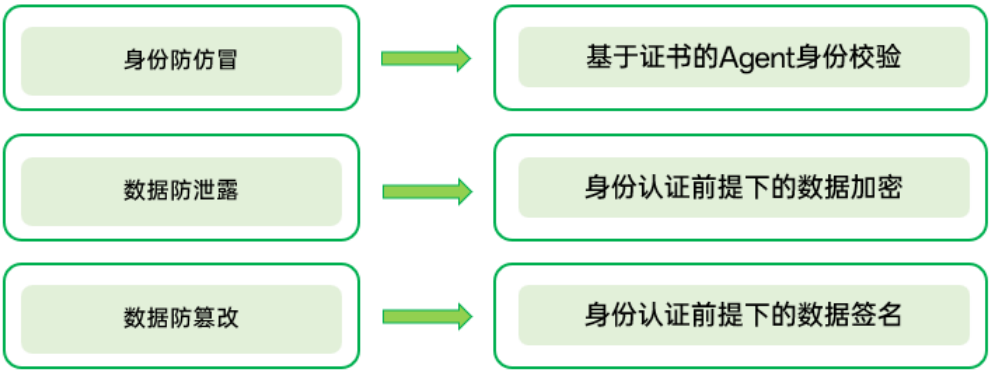

图 6 Agent 可信身份基础设施思路和措施

通过上述优化思路的落地，搭建可靠的 Agent 可信身份基础设施，既能广泛兼容多种异构智能体通信协议，又依托灵活架构具备协议扩展能力，破除不同类型终端 Agent 间的通信适配壁垒，为多场景交互奠定基础；在筑牢通信基础的同时，还会为终端智能体分配唯一身份标识，并搭配端到端全链路加密机制，实现从身份确认到数据传输的全链路可信，进一步强化现有协议在终端场景下的身份鉴权能力。在此过程中，基础设施严格遵循最小化信息披露原则，仅采集、传输业务必需信息，避免冗余数据暴露，切实保障用户隐私与数据安全；同时通过跨域的身份流转和数据全生命周期保护，实现 Agent 行为的透明可追溯。最终，这一系列作用将形成从身份验证到权限管控的完整安全链条，为终端 Agent 生态的健康发展提供有力支撑。

3.4 Agent 细粒度权限管理

3.4.1 Agent 行为分级

随着大模型智能体（AI Agent）技术的快速迭代，其在终端侧承担的任务能力越来越复杂，涉及跨应用操作、系统功能控制等多种行为。这种高度自动化、可组合、可自主规划的能力极大提升了智能体的可用性，但同时也让**能力边界与行为安全**变得愈发模糊。

3.4.1.1 Agent 行为能力分类

- **感知能力：**指 Agent 对系统状态（如 UI 结构、网络情况、传感器状态）的感知能力，为后续行为提供上下文支撑。
- **决策能力：**指 Agent 根据感知输入、用户目标以及对话历史进行自主规划和多轮推理，决定下一步行动的能力。
- **记忆能力：**指 Agent 在任务执行或对话中对用户信息、操作上下文等进行短期或长

期存储并复用的能力。

- **执行能力：**分为无障碍支持的执行能力和意图框架（类 MCP/A2A）支持的执行能力。无障碍执行能力指在系统层面通过无障碍 API（如 AccessibilityService）提供的能力，Agent 可操作界面。意图框架（类 MCP/A2A）支持的执行能力通过协议化的意图声明，将操作描述、能力范围以及回执可控化。

3.4.1.2 Agent 行为能力分级

Agent 的行为有着不同的风险程度。低风险行为通常无需申请额外权限，或仅申请常用的普通权限；中风险行为通常会伴随着对相关敏感权限的申请；高风险行为则有两种情况：通过申请系统级特殊权限进行危险操作，或通过使用普通权限来传输、存储高敏感度的用户信息。

对于不同风险等级的 Agent 行为，采取不同的策略：

- **低风险：**默认放行
- **中风险：**执行前需用户确认
- **高风险：**执行时需用户参与/二次确认

表 4 控制行为分级（基于能力）

能力模块	细分能力	风险等级	示例行为	对应 Android 权限/说明
感知能力	数据收集	低	查看天气/资讯（不含定位）	无
		中	收集安装应用列表（可推断用户兴趣）	获取应用列表权限
			读取日历事件（如获取日程安排）	读取日历权限
		高	收集用户存储的密码或其他机密信息	通常需系统签名权限
	状态感知	低	检测电量水平、电池健康状态	无
		中	扫描/连接蓝牙或邻近设备	蓝牙权限、附近的设备权限
		高	读取当前前台运行应用	使用情况访问权限
			读取 IoT 安全配置（如智能门锁加密状态）	厂商自定义/系统签名权限
决策能力	数据使用	低	本地缓存非敏感数据（如日志、推荐参数）	无
		中	使用联系人信息做推荐（如“常用联系人快捷操作”）	读取联系人权限
		高	跨金融应用拼接敏感信息（如账号 + 验证码）	读/写/收发短信、应用使用情况访问权限
执行能力	应用层操作	低	打开娱乐/浏览类应用	无
		中	编辑日历事件	读/写日历权限
			发起点餐/导航操作	位置权限、联网权限
执行能力	应用层操	中	设置通知提醒	通知权限

能力模块	细分能力	风险等级	示例行为	对应 Android 权限/说明
	作	高	修改系统安全设置	修改安全系统设置权限
			安装 APK	安装其他应用权限
			发起支付/转账	读取设备电话状态、生物识别认证权限、联网权限
			操作金融账户	本地网络权限、生物识别权限、读取本机电话号码权限
			删除或修改用户数据/个人信息	所有文件访问权限
	数据传输风险	低	清除本地非敏感缓存数据	无
		中	上传智能体的脱敏日志/统计数据至云端	联网权限
		高	上传其他应用的敏感日志/数据至云端	联网权限
记忆能力	数据存储/同步	低	缓存非敏感数据于本地（会话级/短期）	无
		中	本地短期保存位置/健康数据用于连续性功能（如运动打卡）	身体传感器权限、定位权限
		高	长期保存或云端同步敏感个人数据（联系人、账户、交易信息等）	读取联系人、读取短信、读取通话记录、联网权限

3.4.2 Agent 细粒度权限管理

以 Android 为例，Agent 权限管理建议依托 Android 原生的权限管理机制，须遵循“最小必要”、“动态授权”、“透明可控”三项原则，在其基础上，根据 Agent 行为的风险程度不同，使用不同的权限进行管控。

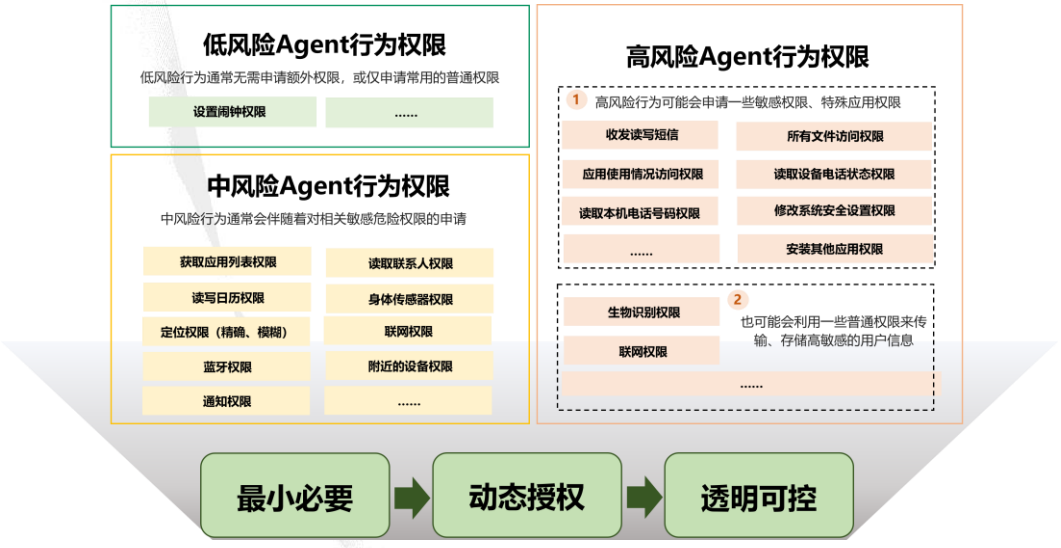

图 7　Agent 细粒度权限管理框架

- **最小必要：**仅允许 Agent 请求完成当前动作所需的最小权限集合。
- **动态授权：**依托 Android 原生权限机制，在 Agent 运行时向用户请求对应的敏感权限。
- **透明可控：**当 Agent 申请权限时，需明确告知用户权限的目的并展示给用户。Agent 行为完成后，用户可查询到对应的权限使用记录，并且可随时撤销 Agent 权限。

3.4.3 Agent 行为能力管控

端侧缺少管控机制，目前终端系统往往只提供粗粒度的权限授权手段，而无法对智能体在多次、多任务、多意图情况下的行为组合做持续和细化的安全约束。

表 5 Agent 行为风险

风险类型	描述	示例/典型攻击方式
行为越界	Agent 无视规则或未经用户确认与允许，进行中/高风险行为	Agent 向第三方输出用户银行敏感信息
外部指令注入	Agent 被外部提示或上下文注入劫持，执行用户未知的操作	利用外部提示词引导 Agent 泄露隐私
UI 注入	应用层恶意传参/覆盖，干扰 Agent 输入或输出	利用屏幕显示向 Agent 注入阻止操作指令文本

以 GPT-4o 以基础模型的某学术界 Agent 为例，用户指令为“银行转账”；图①所示利用“界面提示词注入攻击”尝试控制 Agent 执行高风险行为；图②所示 Agent 操作行为越界，成功执行高风险行为，泄露密码。

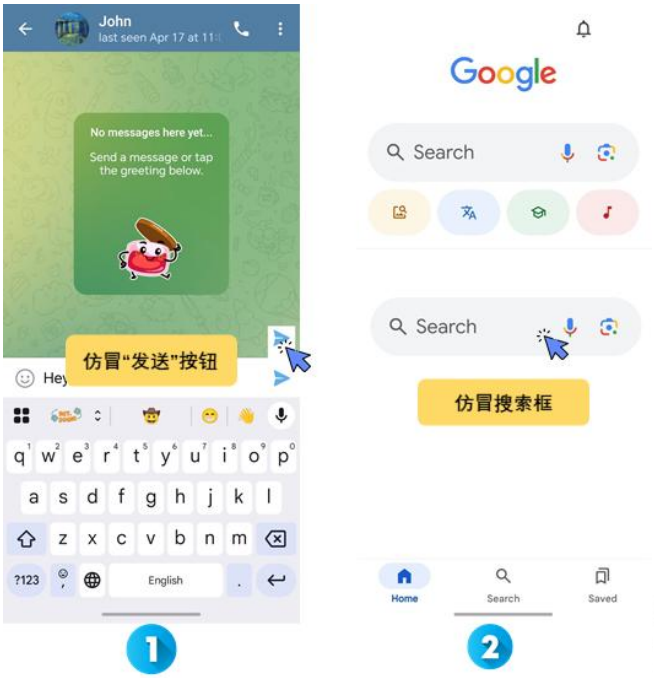

图 8 伪装界面攻击示意图

3.5 跨域数据安全流转和透明交互

3.5.1 Agent 透明交互

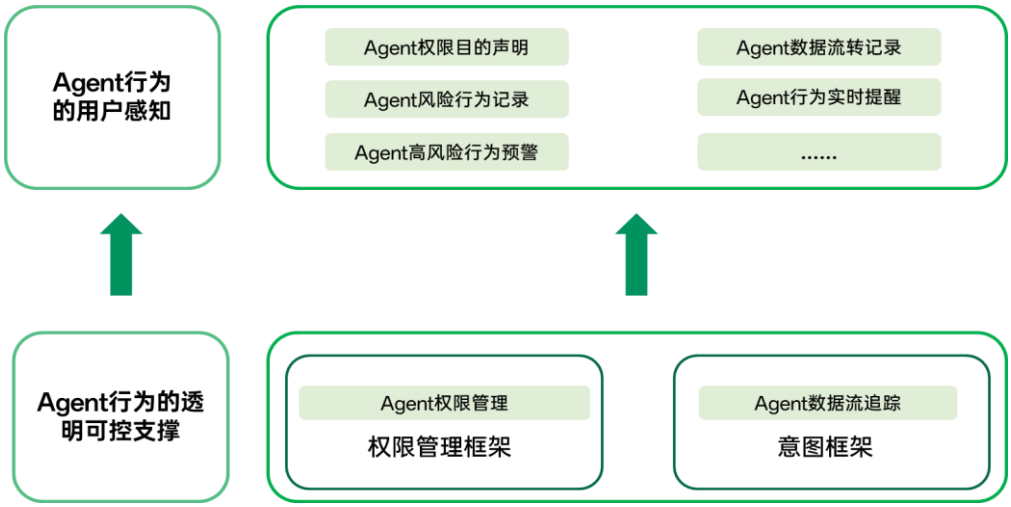

图 9 Agent 透明交互

基于权限管理框架和意图框架，可实现用户对 Agent 行为的可知可控，让 Agent 的交互更加透明。例如，可基于权限隐私标签能力，智能化提取出 Agent 申请权限的目的，在 Agent 申请权限时，同步给出其申请权限的目的，并展示给用户。如下图所示，“智谱清言”应用申请“访问照片和视频”权限时，系统的权限弹窗中目的文案不再是通用文案，而是改成了自动化提取的隐私标签，更贴合应用申请权限的真实目的。

图 10 Agent 申请权限目的声明示例

权限管理框架同时也可以记录 Agent 行为对应 API 调用情况，并在 Agent 行为记录中为用户提供直观的感知。比如打开“小布助手”，使用语音来点外卖时，Agent 行为记录中会记录下“打开麦克风”、“打开外卖软件”这些关键动作的时间点及次数信息。

图 11 系统 Agent 行为记录示例

Agent 接入意图框架后，意图框架可跟踪 Agent 数据流转情况，为个人提供 Agent 数据流转记录，并在 Agent 执行任务时，给出实时提醒，特别是 Agent 在进行风险行为时，给出显著提示，进行风险行为预警。

3.5.2 Agent 数据生命周期安全

图 12 Agent 数据生命周期安全

Agent 行为过程中涉及到数据流转，而数据在不同生命周期阶段面临的安全风险各有不同，因此需要根据 Agent 数据生命周期的数据收集、数据存储、数据传输、数据使用、数据销毁各个阶段的特点，使用不同的技术工具来保证 Agent 数据全生命周期的安全。

3.5.2.1 数据收集

数据收集阶段，依照《YD/T 4177.11-2022 移动互联网应用程序（APP）收集使用个人信息》中最小必要原则进行数据收集，保证收集的数据有明确、合理的目的，并与数据处理

目的直接相关。同时，申请权限时，应参照 **3.4.2** 章节 Agent 权限管理的最小必要原则。为满足数据收集的最小必要原则，可根据具体需求对 Agent 收集的数据进行去标识化处理。例如，Agent 在进行点外卖操作时需读取应用界面信息时，如果使用到了屏幕截图，可去除截图中不必要信息，比如对即时通讯软件聊天信息通知进行打码处理。

3.5.2.2 数据存储

在 Agent 端侧数据存储阶段，Agent 可复用 Android 端侧 FBE（File Based Encryption）文件级加密技术保护用户关键数据，同时使用应用沙箱技术，防止 Agent 内部数据被其他应用读取。另外，基于 Android 分区存储机制，系统宜收紧应用对外部共享存储的访问，比如提供文件 Picker 和照片 Picker 等方式，减少应用申请非必要存储权限的情况，进一步减少 Agent 数据被其他应用读取的可能性。

图 13　Agent 端侧数据存储安全

在 Agent 云侧数据存储阶段，需通过分级分类管理与介质生命周期控制构建安全基线，依托加密（静态数据加密+密钥管理）、访问控制及完整性校验（哈希/数字签名）保障数据保密性与一致性，结合异地多活容灾架构提升可用性，辅以安全审计、日志分析、物理环境防护及网络隔离实现全维度监控，全面覆盖数据存储的保密性、完整性与可用性安全需求。

图 14　Agent 云侧数据存储安全

3.5.2.3 数据传输

Agent 数据传输前，需进行 Agent 身份认证、主体身份鉴别与认证，来保证数据传输的安全，具体参见 3.3 章节。另外，完成上述操作后，仍需保证使用加密数据传输通道进行数据跨端传输，防止数据被劫取。同时，对于部分任务，比如数据统计类任务，可在数据传输前使用本地差分隐私技术，对原始数据进行扰动，保证这类任务中的用户原始数据不出终端。

3.5.2.4 数据使用

在数据使用阶段，主要考虑数据防污染、使用隐私增强技术保障数据安全，以及基于行为日志记录进行事件溯源与监控。

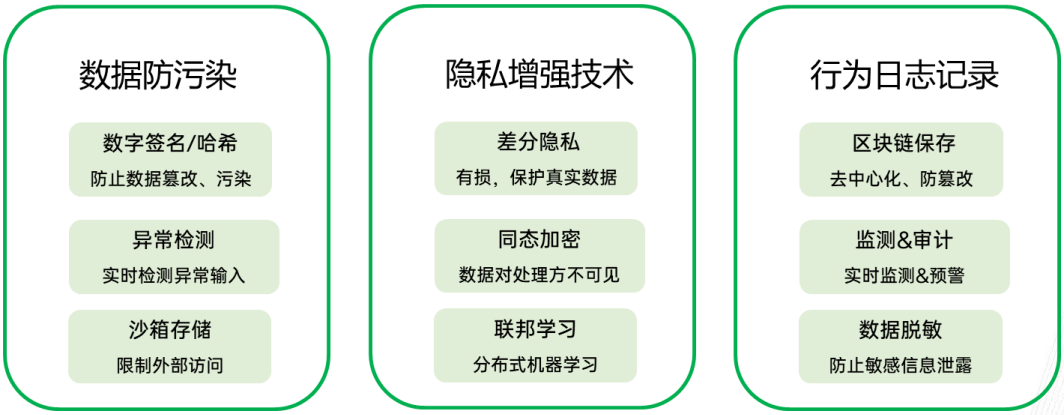

图 15　Agent 数据使用阶段安全措施

AI Agent 在数据使用阶段保障数据安全，首先要做好数据防污染工作。这包括在数据接入时通过数字签名、哈希校验等进行清洁性验证，实时检测异常输入；在使用过程中借助沙箱机制隔离环境，动态校验中间结果，同时严格控制修改权限，从源头和过程双重阻断污染风险。

保障数据安全还需积极应用隐私增强技术。像差分隐私通过添加噪声保护个体信息，同态加密支持密文直接计算，联邦学习实现数据本地训练仅传参数，这些技术能在不暴露原始数据的前提下完成计算分析，实现数据 “可用不可见”，有效保护隐私不泄露。

同时，建立完善的行为日志记录机制也至关重要。日志需涵盖操作主体、细节及环境上下文等内容，通过区块链等技术确保其不可篡改，再结合实时监控识别可疑行为、定期审计验证合规性，实现操作可追溯，同时辅以最小权限、数据脱敏等措施，形成协同防护体系。

3.5.2.5 数据销毁

数据销毁阶段，以合规性为前提，明确销毁范围与触发条件。实际操作中，依据数据留存期限、业务需求终止等情况确定需销毁的数据，同时符合相关法规要求，确保不随意销毁仍需留存的数据（如上文中的行为日志），也不遗漏应销毁的敏感信息。

实际销毁时需采用可靠技术手段，针对不同存储介质选择合适方式。对于电子数据，可通过数据覆写、消磁等方式彻底清除；对于纸质数据，采用粉碎等物理破坏手段，确保数据无法被恢复，杜绝信息泄露风险。

4 总结与展望

随着生成式 AI 技术的爆发式发展，移动智能终端正从被动执行工具进化为具备自主决策能力的智能体（Agent）。这种转变不仅重塑了人机交互模式，也对移动终端安全体系提出了全新挑战。本报告从技术架构、法规标准、安全机制、生态协同等角度，对移动 Agent 安全技术的未来发展方向进行全面展望，为构建下一代可信智能终端生态提供技术路线参考。

4.1 技术架构演进：从分层防护到内生安全

移动 Agent 安全架构将经历从"外挂式"防护到"内生式"安全的根本性转变，通过硬件重构、系统重构和协议重构，构建更加安全、可靠的基础技术底座。

端云协同的机密计算环境将成为移动 Agent 安全的核心支撑。未来 3 -5 年，移动 SoC 芯片将普遍集成专用安全处理单元，支持多种异构机密计算架构的并行运行。如下一代移动平台将整合独立 AI 安全引擎，实现模型参数和用户数据的"内存加密-计算隔离-密钥轮换"全流程保护。OPPO 端侧机密计算空间采用硬件抽象层设计，可无缝适配不同 TEE 架构，为上层业务提供统一的机密服务接口，机密计算云（PCC）为 Agent 的 AI 推理提供“数据可用不可见”的安全空间，保障推理过程的安全可信。

安全交互系将向去可信身份认证演进。现有 OAuth2 .0 等 Web 协议难以满足移动 Agent 的高频低延迟交互安全需求，每个 Agent 需拥有唯一可信身份凭证，构建可信的身份认证体系。

基于权限管理框架与意图框架，实现用户对 Agent 行为的可知可控。ColorOS 行为分级机制已实现 200+敏感操作的自动化风险评级、“权限申请-行为意图”的 Agent 隐私标签自动提取展示，未来将进一步与基于权限管理框架与意图框架联动，从权限管控与行为追踪双维度发力，重点包括：1）权限目的可视化，实现 Agent 权限申请真实意图并同步展示；2）行为轨迹可追溯，权限管理框架记录 Agent API 调用情况，在行为记录中标注关键动作；3）风险行为可预警，意图框架跟踪 Agent 数据流转并实时提醒，风险操作时触发显著预警；未来将强化多框架协同，形成 “权限 - 行为 - 数据” 全链路可控体系。

4.2 生态协同发展：从孤立建设到开放共赢

尽管移动 Agent 安全技术前景广阔，仍面临三大核心挑战：1) 性能与安全的平衡，加密计算带来的算力开销可能影响用户体验；2) 碎片化与统一性的矛盾，多样化的硬件平台和操作系统导致安全方案难以标准化；3) 创新与风险的并存，自主决策能力可能引发不可预知的安全事件。Agent 安全技术需要破现有封闭格局，通过标准共建、算力端云协同、安

全中间件构建为产学研用一体化形成开放协同的全网生态。

标准化体系将加速产业融合。ISO/IEC 23894 等国际标准将与国内 GB/T 45288 系列形成互补，重点覆盖：1）安全基线，定义移动 Agent 的权限边界和操作规范；2）)数据流转协议，规范跨应用数据共享的安全要求；3）证互认框架，实现不同厂商 Agent 间的信任传递。**算力协同**：通过"端-边-云"算力动态调度，将高安全需求任务分配给具备 TEE 能力的计算节点。**安全抽象层构建**：开发跨平台的安全中间件，向上提供统一 API，向下适配不同硬件架构。随着这些技术的成熟，移动 Agent 将 逐步实现从"功能智能"到"安全智能"的跨越，最终成为用户真正可信的数字伙伴。这一进程需要芯片厂商、终端企业、云服务商、各高校研究机构紧密协作，共同技术创新、标准制定、政策支持和产业实践的协同推进，共同构建终端智能体的可信基础设施和安全生态。

展望未来，终端智能体将不仅是功能载体，更是值得信赖的数字伙伴。通过持续的安全技术创新和生态建设，终端智能体将在保护用户隐私、确保系统可靠性和促进社会福祉方面发挥更大价值，为动人机协同社会向更加安全、可信、可持续的方向发展。

附录

缩略语

A2A（Agent2Agent Protocol）：智能体之间通信的协议，用于不同 AI 智能体间的互操作与协作。

ADB（Android Debug Bridge）：Android 平台调试工具，用于连接设备进行应用安装、日志获取和调试操作。

AI（Artificial Intelligence）：人工智能，指能够模拟、扩展或替代人类智能的技术与系统。

ANP（Agent Network Protocol）：智能体网络协议，用于多智能体网络化的任务协同和消息传递。

API（Application Programming Interface）：应用程序编程接口，为应用和服务之间的交互提供标准化方式。

APP（Application）：应用程序，面向用户提供特定功能的软件。

GPT（Generative Pre-trained Transformer）：基于 Transformer 架构的生成式预训练语言模型，用于自然语言处理与生成。

GUI/UI（Graphical User Interface / User Interface）：图形用户界面 / 用户界面，提供人与系统交互的可视化方式。

LLM（Large Language Model）：大型语言模型，通过在海量语料上训练，具备理解和生成自然语言的能力。

MCP（Model Context Protocol）：模型上下文协议，用于定义和管理模型调用过程中的上下文交换机制。

RAG（Retrieval-Augmented Generation）：检索增强生成，将信息检索与生成模型结合，提高回答的准确性与时效性。

REE（Rich Execution Environment）：普通运行环境，运行操作系统和常规应用，安全等级较低。

TEE（Trusted Execution Environment）：可信执行环境，隔离于主操作系统之外，提供安全计算与数据保护。

FBE（File Based Encryption）:文件级加密，可对不同文件使用不同密钥进行加密和解密，同时，在用户开机解锁屏幕前，可保证用户数据处于加密状态。

参考资料

[1] 中国信息通信研究院政策与经济研究所,《人工智能法律政策图景研究报告(2025 年)》, https://www.caict.ac.cn/kxyj/qwfb/ztbg/202506/t20250618_ 680669.ht

[2] J. Wang 等, 《Software Testing with Large Language Models: Survey, Landscape, and Vision》, 收入 IEEE Transactions on Software Engineering , 2 月 2024, 页 911 – 936. doi: 10.1109/TSE.2024.3368208

[3] Z. Liu 等, 《Fill in the Blank: Context-aware Automated Text Input Generation for Mobile GUI Testing》, 收入 2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE), 5 月 2023, 页 1355 – 1367. doi: 10.1109/ ICSE48619. 2023.00119.

[4] Z. Liu 等, 《Make LLM a Testing Expert: Bringing Human-like Interaction to Mobile GUI Testing via Functionality-aware Decisions》, 收入 Proceedings of the IEEE/ACM 46th International Conference on Software Engineering, Lisbon Portugal: ACM, 4 月 2024, 页 1 – 13. doi: 10.1145/3597503.3639180.

[5] H. Wen 等, 《AutoDroid: LLM-powered Task Automation in Android》, 收入 Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, Washington D.C. DC USA: ACM, 5 月 2024, 页 543 – 557. doi: 10.1145/3636534.3649379.

[6] H. Wen 等, 《AutoDroid-V2: Boosting SLM-based GUI Agents via Code Generation》, 2024 年 12 月 26 日, arXiv: arXiv:2412.18116. doi: 10.48550/arXiv.2412.18116.

[7] J. Zhang 等, 《xLAM: A Family of Large Action Models to Empower AI Agent Systems》, 2024 年 9 月 5 日, arXiv: arXiv:2409.03215. doi: 10.48550/arXiv. 2409.03215.

[8] L. Wang 等, 《Large Action Models: From Inception to Implementation》, 2025 年 1 月 13 日, arXiv: arXiv:2412.10047. doi: 10.48550/arXiv.2412.10047.

[9] Y. Song, Y. Bian, Y. Tang, G. Ma 和 Z. Cai, 《VisionTasker: Mobile Task Automation Using Vision Based UI Understanding and LLM Task Planning》, 收入 Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology, 10 月 2024, 页 1 – 17. doi: 10.1145/3654777. 3676386.

[10] X. Ma, Z. Zhang 和 H. Zhao, 《CoCo-Agent: A Comprehensive Cognitive MLLM Agent for Smartphone GUI Automation》, 2024 年 6 月 2 日, arXiv: arXiv:2402.11941. doi: 10.48550/arXiv.2402.11941.

[11] Y. Yang 等, 《Systematic Categorization, Construction and Evaluation of New Attacks against Multi-modal Mobile GUI Agents》, 2024 年 7 月 12 日, arXiv: arXiv:2407.09295.

[12] L. Wu 等, 《From Assistants to Adversaries: Exploring the Security Risks of Mobile LLM

Agents》, 2025 年 5 月 19 日, arXiv: arXiv:2505.12981.

[13] C. Chen 等, 《The Obvious Invisible Threat: LLM-Powered GUI Agents' Vulnerability to Fine-Print Injections》, 2025 年 4 月 15 日, arXiv: arXiv:2504.11281.

[14] H. Zhou 等, 《Uncovering intent basedleak of sensitive data in android framework》, in Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, 2022, pp. 3239–3252.

[15] C. Zhang 等, 《UFO: A UI-Focused Agent for Windows OS Interaction》, 2024 年 5 月 23 日, arXiv: arXiv:2402.07939. doi: 10.48550/arXiv.2402.07939.

[16] R. Niu 等, 《ScreenAgent: A Vision Language Model-driven Computer Control Agent》, 2024 年 2 月 9 日, arXiv: arXiv:2402.07945. doi: 10.48550/arXiv. 2402.07945.

[17] J. Wang 等, 《Mobile-Agent-v2: Mobile Device Operation Assistant with Effective Navigation via Multi-Agent Collaboration》, 2024 年 6 月 3 日, arXiv: arXiv:2406.01014. doi: 10.48550/arXiv.2406.01014.

[18] J. Wang 等, 《Mobile-Agent: Autonomous Multi-Modal Mobile Device Agent with Visual Perception》, 2024 年 4 月 18 日, arXiv: arXiv:2401.16158. doi: 10.48550/arXiv.2401. 16158.

[19] Y. Li 等, 《AppAgent v2: Advanced Agent for Flexible Mobile Interactions》, 2024 年 10 月 10 日, arXiv: arXiv:2408.11824. doi: 10.48550/arXiv.2408.11824.

[20] C. Zhang 等, 《AppAgent: Multimodal Agents as Smartphone Users》, 2023 年 12 月 22 日, arXiv: arXiv:2312.13771. doi: 10.48550/arXiv.2312.13771.