CAI × ERC-8004 Security Framework

Demo 演讲稿分镜 – ETH Shanghai 2025

总时长: 2分50秒 (170秒)
格式: 中文演讲 + 简体字幕
演讲者: 项目负责人
背景音乐: 轻快科技感音乐 (低音量)

🎬 第一部分：问题场景 (0:00 – 0:20，20秒)

画面

- **镜头 1 (0:00–0:05)**: 黑屏淡入，显示项目 Logo
  - 文字动画："CAI × ERC-8004 Security Framework"
  - 副标题："A Verifiable AI-Agent Commerce Protocol"
- **镜头 2 (0:05–0:15)**: 问题场景动画
  - 左侧：AI Agent 自动购物的流程图
  - 右侧：红色警告标志（欺诈、篡改、身份冒用）
  - 数据可视化：每年因 AI 欺诈损失 $XXX 亿
- **镜头 3 (0:15–0:20)**: 过渡到解决方案
  - 画面：盾牌图标 + 区块链网络图
  - 文字："我们的解决方案"

台词

【0:00–0:05】
大家好，我们是 CAI 团队。
（停顿 1 秒）

【0:05–0:15】
随着 AI Agent 获得自主交易能力，一个严峻的问题出现了：
如何防止 AI 代理欺诈、数据篡改和身份冒用?
现有的支付系统缺乏端到端的可验证性。

【0:15–0:20】
我们的答案是：CAI 框架 —— 基于 ERC-8004 标准的
可验证 AI 商业协议。

字幕

0:05 – 问题：AI Agent 自主交易的安全风险
0:10 – 欺诈 | 篡改 | 身份冒用
0:15 – 解决方案：CAI × ERC-8004 Framework

🎯 第二部分：解决方案概览 (0:20 – 0:40，20秒)

画面

- **镜头 4 (0:20–0:30)**: 架构图动画
  - 四层防护：
    1. DID 身份层（CAIRegistry）
    2. 授权凭证层（Mandate VC）
    3. 防篡改链（AHIN Hash Chain）
    4. 链上锚定（Ethereum）
  - 每层逐个淡入，带动画效果
- **镜头 5 (0:30–0:40)**: 核心特性卡片
  - 三张卡片翻转动画：
    - 🔐 强身份验证
    - 🛡️ 防篡改证明链
    - 🔍 端到端可审计

台词

【0:20–0:30】
CAI 框架提供四层安全防护：
第一，去中心化身份注册表，确保 Agent 真实性；
第二，可验证凭证系统，精确控制授权范围；
第三，哈希链锚定，防止任何数据篡改；
第四，定期同步到以太坊主网，实现不可抵赖。

【0:30–0:40】
三大核心特性：
强身份验证、防篡改证明链、端到端可审计。
每一笔交易都有完整的验证路径。

字幕

0:20 – 四层防护架构
0:25 – CAIRegistry | Mandate VC | AHIN Chain | Ethereum
0:30 – 核心特性
0:35 – 验证 | 防篡改 | 可审计

💻 第三部分：核心演示 (0:40 – 2:20，100秒)

画面分镜

镜头 6：用户授权 (0:40–0:55，15秒)

- **画面**：
  - 前端界面：用户登录（MetaMask 连接动画）
  - 表单：授权设置（预算 100 DAI，有效期 24 小时）
  - 点击"签发 Mandate VC"按钮
  - 成功提示：绿色勾选 + 凭证哈希显示
- **代码框**（右侧小窗口）：

javascript

```
// 签发授权凭证
const mandate = {
  agent: "did:ethr:0xABCD...",
  budget: "100 DAI",
  expiry: "24h",
  whitelist: ["merchant1.eth"]
};
const vcHash = await issueVC(mandate);
```

镜头 7：AI Agent 创建购物车 (0:55–1:15，20秒)

- **画面**：
  - 切换到 Agent 控制台（黑色终端风格）
  - 日志输出：

[Agent] Scanning mandate...
[Agent] Budget: 100 DAI ✓
[Agent] Creating cart...
[Agent] Item 1: AI Training Dataset – 50 DAI
[Agent] Item 2: GPU Hours – 45 DAI
[Agent] Total: 95 DAI ✓ (Within budget)
[Agent] Generating Cart VC...
[Agent] Cart Hash: 0x7f3e...

- 右侧：Cart VC JSON 结构可视化
- **动画效果**：
  - 购物车项目逐行淡入
  - 预算检查（绿色进度条从 95/100）
  - Cart Hash 生成动画（哈希值滚动后锁定）

镜头 8：链上交易初始化 (1:15–1:40，25秒)

- **画面**：
  - 前端：交易卡片
    - 左侧：交易详情（Agent、商户、金额、Cart Hash）
    - 右侧：区块链浏览器实时视图（Sepolia Testnet）
  - 按钮："初始化交易"（带加载动画）
  - 交易广播：
    - Pending → Confirmed (3/6 区块确认)
    - 显示 Transaction ID
- **合约调用可视化**（底部时间轴）：

User → ERC8004Agent.initiateTransaction()
↓
[验证 Agent DID]
↓
[验证 Mandate VC]
↓
[生成 Transaction ID]
↓
[触发 TransactionInitiated 事件]

- **Etherscan 截图**（右上角小窗口）：
  - 显示真实的交易哈希
  - 状态：Success
  - Gas Used: 128,456

镜头 9：支付完成与收据 (1:40–2:00，20秒)

- **画面**：
  - 支付网关界面（模拟 PayPal 风格）
  - 支付流程：

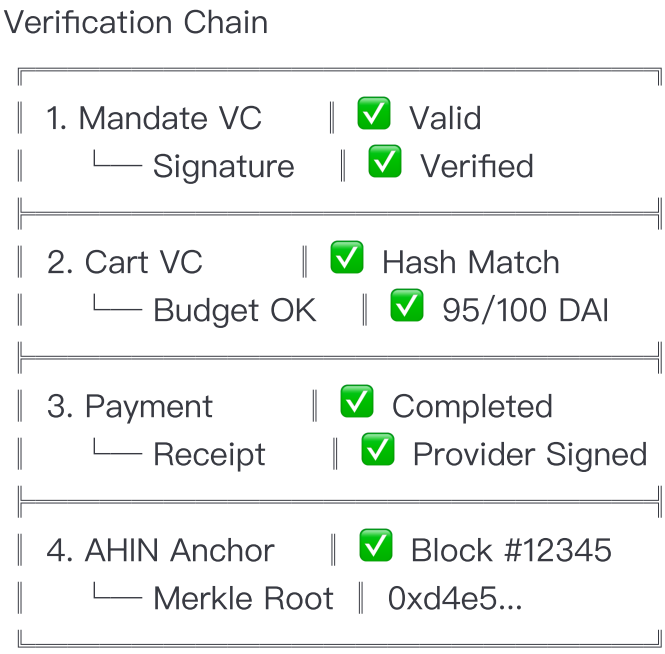
Payment Gateway

```
┌──────────────────────────────────────┐
│ Amount: 95 DAI                       │
│ Cart Hash: 0x7f3e...                 │
│ Status: COMPLETED                    │
│ Receipt: 0x9a2b...                   │
└──────────────────────────────────────┘
```

- 支付提供商签名验证（绿色勾选）
- Agent 调用 `completeTransaction()`
- 交易状态更新：Pending → Completed
- **动画效果**：
  - 支付流程进度条（100%）
  - Receipt Hash 与 Cart Hash 配对验证动画（两个哈希值连线）

镜头 10：审计仪表盘 (2:00–2:20，20秒)

- **画面**：
  - 切换到审计仪表盘（完整界面）
  - **左侧**：证明链可视化



- **右侧**：
  - Agent 信誉评分（ChainRank: 95/100）
  - 历史交易图表（成功率 98%）
  - "下载审计报告"按钮（点击后生成 JSON）
- **最终动画**：
  - 整个证明链从上到下"扫描"动画（绿色光效）
  - 显示"Fully Verified ✓"徽章

台词

【0:40–0:55】
现在进入核心演示。
首先，用户通过前端签发一个 Mandate VC，
设定预算 100 DAI，有效期 24 小时。
这个凭证被签名并存储在 CAIRegistry 合约中。

【0:55–1:15】
接下来，AI Agent 读取授权，创建购物车。
它选择了两个商品：AI 训练数据集 50 DAI，GPU 算力 45 DAI，
总计 95 DAI，在预算范围内。
Agent 生成 Cart VC 并计算 Cart Hash。

【1:15–1:40】
然后，Agent 在链上初始化交易。
合约验证 Agent 的 DID 身份，检查 Mandate VC 的有效性，
并生成唯一的 Transaction ID。
您可以看到，这笔交易已经在 Sepolia 测试网成功确认。

【1:40–2:00】
支付完成后，支付网关返回签名的 Receipt。
Agent 调用 completeTransaction 方法，
合约验证 Cart Hash 与 Receipt 的绑定关系。
交易状态从 Pending 变为 Completed。

【2:00–2:20】
最后，在审计仪表盘中，我们可以看到完整的验证链：
从 Mandate 到 Cart、到 Payment、再到链上锚定，
每一步都有密码学证明。
任意第三方都可以验证这条交易的完整性。
右侧显示 Agent 的信誉评分和历史记录。

字幕

0:40 – 步骤 1：用户签发授权凭证
0:50 – Mandate VC: 预算 100 DAI, 24h
0:55 – 步骤 2：Agent 创建购物车
1:05 – Cart Hash: 0x7f3e...
1:15 – 步骤 3：链上交易初始化
1:25 – Transaction ID: 0x1a2b...
1:40 – 步骤 4：支付完成
1:50 – Receipt 已验证 ✓
2:00 – 步骤 5：审计验证
2:10 – 完整证明链 | ChainRank: 95

🚀 第四部分：技术亮点 (2:20 – 2:40，20秒)

画面

- **镜头 11 (2:20–2:30)**: 技术特性动画
  - 三列并排显示：
    - **列 1**：端到端可验证
      - Icon: 🔍
      - 文字："签名 + 哈希链 + 链上锚定"
    - **列 2**：隐私保护
      - Icon: 🔒
      - 文字："ZKP 选择性披露"
    - **列 3**：开放集成
      - Icon: 🔗
      - 文字："GitHub Actions CI/CD"
- **镜头 12 (2:30–2:40)**: 代码库展示
  - GitHub 仓库界面：
    - Stars: 128
    - Forks: 34
    - Issues: 5 open
  - 目录结构快速滚动
  - 测试覆盖率徽章：93.7%
  - License: MIT

台词

【2:20–2:30】
技术亮点：
第一，端到端可验证 —— 决策签名加合约验签，确保不可篡改；
第二，隐私保护 —— 支持零知识证明，只暴露必要信息；
第三，开放集成 —— 我们提供完整的 CI/CD 脚本，
任何团队都可以 5 分钟内部署自己的验证节点。

【2:30–2:40】
我们的代码已在 GitHub 完全开源，MIT 协议。
测试覆盖率 93.7%，包含 6 种攻击场景的防御验证。

字幕

2:20 – 技术亮点
2:25 – 可验证 | 隐私 | 开放
2:30 – GitHub 开源
2:35 – 测试覆盖率 93.7%

🎯 第五部分：未来规划与结尾 (2:40 – 2:50，10秒)

画面

- **镜头 13 (2:40–2:45)**: Roadmap 时间轴
  - Q4 2025: ZK-SNARK 集成
  - Q1 2026: 去中心化风险预言机
  - Q2 2026: 主网部署
- **镜头 14 (2:45–2:50)**: 结束画面
  - 团队 Logo + 联系方式
  - QR Code（GitHub 仓库）
  - 文字："Thank You! | 谢谢！"
  - 背景：缓慢上升的代码雨动画

台词

【2:40–2:45】
下一步，我们计划集成 ZK-SNARK 零知识证明，
并建立去中心化的风险评估网络。

【2:45–2:50】
感谢评审！我们现在可以现场演示或回答问题。
欢迎访问我们的 GitHub 仓库。

字幕

2:40 – Roadmap
2:42 – ZK-SNARK | 去中心化预言机 | 主网
2:45 – 谢谢!
2:48 – GitHub: github.com/cai-framework

📋 拍摄清单

前期准备

- [ ] 部署合约到 Sepolia 测试网
- [ ] 准备 3 个测试账户（用户、Agent、商户）
- [ ] 录制真实交易的屏幕操作
- [ ] 准备审计仪表盘演示数据
- [ ] 测试所有动画效果

拍摄设备

- [ ] 高清屏幕录制软件（OBS Studio / QuickTime）
- [ ] 麦克风（清晰人声）
- [ ] 视频编辑软件（Final Cut Pro / Premiere Pro）
- [ ] 字幕工具（Arctime / Aegisub）

后期制作

- [ ] 添加背景音乐（音量 -20dB）
- [ ] 嵌入简体中文字幕
- [ ] 过渡动画（淡入淡出，0.5秒）
- [ ] 颜色校正（统一色调）
- [ ] 输出格式：MP4, 1080p, H.264

🎤 现场演示备用方案

如果现场演示出现问题

Plan B: 录屏演示

- 播放提前录制的 2 分 50 秒视频
- 演讲者同步讲解（可适当停顿指出关键点）

Plan C: 静态 PPT

- 准备 15 张关键截图 PPT
- 快速翻页配合口头讲解
- 时间控制在 3 分钟内

常见问题准备（Q&A）

Q1: 如何处理 Agent 私钥安全？ A: 我们支持 MPC 多方计算托管，私钥永不暴露在单点。

Q2: Gas 成本如何？ A: 单次锚定 < 50k gas，批量验证降低 80% 成本。

Q3: 与现有支付系统兼容吗？ A: 完全兼容 PayPal、Stripe、稳定币网关，无缝集成。

Q4: 如何防止 Agent 恶意行为？ A: 三层防护：授权限额、实时监控、链上仲裁机制。

📊 演讲节奏控制表

时间段	内容	重点	语速
0:00–0:20	问题场景	吸引注意力	快速
0:20–0:40	解决方案	建立信任	中等
0:40–2:20	核心演示	展示能力	稳定
2:20–2:40	技术亮点	强调创新	快速
2:40–2:50	未来规划	留下印象	坚定

总字数: 约 650 字
平均语速: 230 字/分钟（正常演讲速度）
建议练习: 至少 5 遍，确保流畅度和时间控制

✅ 最终检查清单

- [ ] 视频时长 ⩽ 3 分钟（严格控制）
- [ ] 中文配音清晰无杂音
- [ ] 简体中文字幕完整
- [ ] 所有演示功能可正常运行
- [ ] 合约地址在视频中可见
- [ ] GitHub 仓库链接清晰展示
- [ ] 输出文件名：`demo-video.mp4`
- [ ] 文件大小 < 100MB（便于上传）

制作完成后请保存到: `/demo/demo-video.mp4`
分镜脚本保存到: `/demo/demo-script.txt`（本文件）