arXiv:2603.26970v1 [cs.CR] 27 Mar 2026

HFIPay：面向人性化标识符的隐私保护型跨链加密货币支付

Jian Sheng Wang
Yeah LLC
jason@yeah.app

2026年3月27日

摘要

向电子邮件地址或电话号码发送加密货币，本应像银行转账一样简单；然而，朴素方案将标识符直接映射到区块链地址，使任何知道该标识符的人都能看到接收方的余额和交易历史。

HFIPay 将隐私路由、发送方报价验证和链上申领授权分离。中继在链下解析人性化标识符，仅提交逐意图盲化绑定 ρ_i 以及报价支付元组；区块链既看不到标识符，也看不到可重复使用的收件人标签。在报价验证部署中，中继会返回发送方可验证的链下证明，将 ρ_i 与经证明的绑定密钥承诺关联起来，因此中继无法在资金注入前替换为其他接收方。接收方申领时，通过 ZK- ACE [2]以零知识方式证明：已注资意图的盲化绑定与从同一确定性身份导出的句柄相匹配，从而授权将报价资产和金额发放至选定目的地。

我们形式化定义了两个隐私目标：枚举抗性 和认领前不可链接性，并区分基线部署（信任中继以确保绑定正确性）与经验证报价部署（无需公共注册表，发送方即可验证绑定）。与 n-VM 运行时组合后，同一机制可扩展至跨链结算。由此形成一种由中继辅助但非托管的架构：中继是隐私与可用性依赖，但无法转移资金。

关键词：基于标识符的支付、保护隐私、可验证报价、盲化申领绑定、零知识授权、跨链

1 引言

1.1 动机

加密货币普及受到一个根本性用户体验障碍的阻碍：用户必须管理难以理解的十六进制地址、选择正确的网络，并持有用于支付交易费用的原生代币。这些要求使点对点加密货币支付对非技术用户而言难以使用，即使对有经验的用户来说也容易出错。

相比之下，电子邮件地址、电话号码和社交媒体账号（例如 X/Twitter 用户名）等人性化标识符无处不在，并且已经成为在线身份的主要锚点。如果加密货币可以发送至此类标识符——无需设置钱包、无需复制地址、无需获取 Gas 代币——那么最终的体验将与传统数字支付别无二致。

1.2 隐私挑战

一种简单直接的实现方式是通过确定性函数将用户标识符直接映射到区块链地址（例如 keccak256(`identifier`) $\rightarrow$ `address`），其中标识符可以是电子邮件、电话号码或社交媒体账号。这种方式虽然简单，却会带来严重的隐私漏洞：任何知道用户标识符的第三方，都能推导出其链上地址，并查看其全部交易历史、代币余额以及交易对手关系。

对公众人物而言，这尤其危险。只需知道 `alice@example.com`、公开的电话号码或知名的 X 账号，对手就能还原 Alice 在每条支持的区块链上的完整财务档案。

1.3 我们的方法

HFIPay 通过将标识符路由、接收方绑定、发送方报价验证和领取授权彼此分离，解决了这一问题。发送方只需指定一个便于理解的标识符，例如电子邮件地址。中继会私下将该标识符解析为已注册收款方记录，随机生成一个 `intentId`，推导出一个一次性存款地址，并仅将意图专属盲化绑定 ρ_i 与报价支付元数据一同提交到链上。在经验证报价部署中，中继还必须向发送方提供经证明的报价，以证明报价的 ρ_i 与预期接收方的隐藏绑定句柄相关联。接收方之后可通过证明其控制着与该盲化值绑定的确定性身份来领取款项。

其关键特性包括：

1. **枚举抗性**。 链上数据不会泄露哪些已注资意图属于已知的电子邮件、电话号码或社交账号。2. **申领前不可关联性**。 向同一标识符进行多笔认领前支付，不会暴露可重复使用的公开收款方标签。3. **报价后领取正确性**。 发送方接受有效报价，且由此产生的意图存储了报价中的盲化绑定元组后，领取授权将通过零知识证明在链上强制执行，而非由中继方自由裁量。4. **跨设备恢复**。 接收方可以使用 Va-Dar 在不同设备上重新推导出相同的确定性身份，无需公开标识符注册表。5. **加密隐匿性**。 在集成客户端流程中，发送方和接收方都无需手动处理特定区块链地址或 Gas。6. **跨链路由**。 结合 n-Vm 架构，同一领取机制即可将资金释放到不同的目标虚拟机上。

简单来说，HFIPay 旨在提供“发送至电子邮件”的用户体验，同时确保标识符解析过程保密，并将支出权限检查转移到区块链上。

1.4 贡献

本文作出以下贡献:

1. **协议设计**。 我们将 HFIPay 定义为一种中继辅助路由层，采用与意图绑定的盲化绑定和报价资产元组，并在资金注入前予以承诺，同时提供可选的发送方可验证报价，从而能够在发送资金前检测接收方替换，并在资金注入后通过链上机制强制执行领取授权（第 3 节）。2. **授权组合**。 我们采用 ZK- ACE [2]实现领取时授权，并展示 Va-Dar [3]如何在不公开枚举标识符的情况下，支持跨设备的接收方私密恢复与发现（第 2、3 节）。3. **隐私分析**。 我们形式化分析链上观察者面临的枚举抗性和认领前不可链接性，并明确说明认领后哪些信息会变得可链接（第 5 节）。4. **绑定模式**。 我们区分基线部署与经验证报价部署；后者使用经认证的链下绑定证明，而非公开标识符注册表（第 2、6 节）。5. **跨链与实现概要**。 我们展示相同的申领关系如何与 n-Vm 运行时组合，并概述兼容以太坊虚拟机和 Solana 的实现方案（第 4、7 节）。

HFIPay 构建于 Ace-Gf 身份框架之上 [1], 使用 ZK-ACE 实现可由区块链验证的授权 [2], 并使用 Va-Dar 实现私密确定性恢复 [3]。其路由机制与隐身地址系统（ERC-5564 [4]）有关联，但在应用层运行，并具有不同的信任边界；详细比较见第 8 节。

2 系统模型

2.1 参与方

该协议涉及五种角色:

定义 1 （发送方）。 通过指定接收方电子邮件地址 e_B、资产标识符 a、金额 v和目标链标识符 c来发起支付的一方 A 。其中， a表示规范源链资产描述符，例如原生 *ETH/SOL*、*ERC-20* 合约地址或 *SPL* 铸币地址。如果启用了退款， A 还会使用相关区块链上的发送方控制的签名凭证或智能账户策略授权退款目的地 γ_A 。

定义 2 (接收方)。 一方 B 控制着一个确定性身份根 REV_B。 B 可从该根推导出一个身份承诺 $\mathsf{ID}_{\mathsf{com}B}$ ，如 ZK-ACE *[2],* 所述，并为每个绑定纪元 e 生成一个私密领取绑定句柄

$$u_{B,e} = H(\mathsf{Derive}(\mathsf{REV}_B, \mathsf{Ctx}_{\mathsf{bind}} \| e)).$$

B 还会推导出相应的链下绑定密钥承诺

$$K_{B,e} = H(\texttt{"hfipay:bind-key"} \| u_{B,e}).$$

部署应从粗粒度的公开纪元计划中选择 e （例如按日或按周轮换的计数器），而不是使用收件人专属随机标签，这样，公开 e作为意图元数据本身就不会生成可重复使用的收件人标签。 B 可以使用 VA-DAR *[3]*在不同设备上加载或恢复相同的根密钥。

定义 3 （中继服务）。 一种服务 $\mathcal{R}$ ，用于维护私有目录

$$\mathsf{Norm}(e) \mapsto (\mathsf{ID}_{\mathsf{com}}, u_e, K_e, e, \texttt{mode}, \texttt{metadata}),$$

以及标识符到待处理意图的私有映射。 $\mathcal{R}$ 代表用户支付区块链交易费用（中继方付费模式）。

定义 4 （绑定层）。 一种可选的应用层或第三方服务 $\mathcal{I}$，用于帮助建立规范化标识符与接收方当前绑定密钥承诺之间的绑定 $K_{B,\ e}$。在基线部署中，此角色可由常规的电子邮件/*OIDC* 入驻流程实现。在经验证报价部署中，它会签发一份证明

$$\tau_B \leftarrow \mathsf{Attest}_{\mathcal{I}}(\texttt{"hfipay:bind-attest"} \| \mathsf{Norm}(e_B) \| K_{B,e} \| e \| T_{\mathsf{bind}}).$$

关键在于， $\mathcal{I}$ 必须独立验证出示方确实控制 $\hat{}e_B$ （例如，自行发送电子邮件一次性密码，或直接验证 *OIDC* 令牌），且绝不能依赖中继对标识符所有权的断言。否则，恶意中继可能会向 $\mathcal{I}$ 出示自己的 K ，同时声称该标识符属于 $\hat{}e_B$，从而使经验证报价部署退化为基线信任模型。

定义 5 (观察者**)**。 任何一方 $\mathcal{O}$ 可以读取公开区块链状态，但无法访问 $\mathcal{R}$ 的私有数据。这包括区块链分析服务、其他用户以及公众。

2.2 注册与绑定模式

在接收本文分析的主协议中的资金之前，接收方需完成一次性注册。令 $\hat{}e_B$ =Nor m(e_B)。接收方根据 ZK-ACE 模型导出身份承诺

$$\mathsf{ID}_{\mathsf{com}B} = H(\mathsf{REV}_B \| s_B \| \mathsf{domain}_{\mathsf{id}})$$

并选择或更新绑定纪元 e。对于该纪元，接收方按上述定义导出 $u_{B,\ e}$ 和 $K_{B,\ e}$ 。当某种部署模式在公开意图元数据中记录纪元标签时， e 应从粗粒度公共计划（例如按日或按周计数）中选取，而非使用收件人专属随机数，这样发布 e 就不会产生可重复使用的收件人标签。

我们考虑两种部署模式:

- **基线部署**。 应用程序会验证对 $\hat{}e_B$ 的控制权（例如，通过电子邮件一次性密码、魔法链接或 OIDC 登录）一次，并存储一个私有目录条目，其中包含 $\hat{}e_B$、$\mathsf{ID}_{\mathsf{com}B}$, $u_{B,\ e}$ 和纪元元数据。此模式提供了简单的用户体验，但初始标识符注册和逐意图收件人绑定仍依赖中继信任假设。

- **经验证报价部署**。 绑定层还会证明规范化后的标识符、当前绑定密钥承诺、绑定纪元以及有效期限。中继会将 $(K_{B,\,e},\ \tau_B)$ 与目录条目一并存储，之后提供一份发送方可验证报价，证明每个报价中的 ρ_i 均源自与经证明的 $K_{B,\,e}$ 相同的隐藏句柄。

Va-Dar 在此处是互补方案，而非替代方案。它让接收方能够在新设备上通过标识符和口令私下恢复相同的 REV_B ，从而重新推导出相同的 $\mathrm{ID}_{\mathsf{com}\,B}$ 以及相关的纪元范围绑定句柄，而无需暴露公开标识符注册表 [3]。

为降低中继数据库日后遭到泄露时发生事后去匿名化的风险，部署时 SHOULD 定期或按需轮换绑定纪元，并且仅在与该纪元关联的未完成意图全部结算后，才保留旧的 $(u_{B,\,e},\ K_{B,\,e})$ 值。公开意图元数据仍可包含恢复所需的粗粒度纪元标签 e_i ；涉及隐私的秘密是 $u_{B,\,e_i}$，而不是纪元标签本身。

以下形式化协议假设采用*已注册接收方模式*：发送方发起支付时，中继已为该标识符存有目录条目。延迟至首次收款时再进行引导注册属于部署扩展，但其初始绑定步骤继承所用注册模式的信任模型。

2.3 威胁模型

我们考虑以下对手能力：

- **被动链上观察者：** $\mathcal{O}$ 可以读取所有链上状态，包括合约存储、交易历史和事件日志。
- **标识符信息：** $\mathcal{O}$ 可能知道特定用户的电子邮件地址、电话号码或社交账号。
- **目录泄露：**对手可能入侵中继的私有目录或意图数据库；我们将在第 5.4 节分析其影响。
- **公开承诺信息：**对手可能从其他场景获知公开身份承诺或目标地址，并试图将其与 HFIPay 声明关联起来。
- **提交抢跑：**通过内存池提交的任何交易都可能被另一方复制并提交。
- **接收方替换尝试：** 恶意中继可能会在发送方为意图注资前，尝试为源自其自身或同伙隐藏句柄的盲化绑定报价。

协议不 防范以下情况：

- 接收方的设备、口令或底层确定性身份根遭到入侵。
- 在初始注册期间发生电子邮件账户接管或 OIDC 账户接管。
- 网络层的流量分析（IP关联）。

- $\mathcal{R}$ 与链上观察者串通，实时暴露目录内容。

- 中继审查或可用性故障。

- 绑定签发方不可用；如果 $\mathcal{I}$ 无法访问，在经过验证的报价模式下，新注册和纪元轮换将无法获得新的证明，受影响接收方的系统会退化为基线信任模型。

- 恶意或遭到入侵的绑定签发方 $\mathcal{I}$提供错误证明。
- 跨链扩展中对外部链存款的验证不正确。
- 在基线部署中，恶意中继替换接收方，而该部署仍处于可信运营方模式。

具体而言，我们重点关注两项属性：

1. 针对仅能看到公链信息且可能知晓人性化标识符的各方的隐私；以及 2. 在经验证报价部署中，发送方接受报价后的声明正确性，即发送方针对某项意图检查了报价绑定、确定性存款地址和已登记的链上意图元组之后的声明正确性。

基线部署保留了应用层假设：资金注入之前，一次性标识符登记和逐意图接收方绑定正确性均须成立。经验证报价部署则取消了后一个假设：发送方验证了证明 τ_B、报价证明、本地重新计算的存款地址和已登记的链上意图元组后，中继便无法诱使其为某项意图注入资金，而该意图的盲化绑定却是根据不同的隐藏句柄导出的，或其报价资产元数据被中继暗中更改。资金注入后，中继仍可延迟处理或审查交易，但若不在资金注入前更改已承诺的盲化值并导致发送方验证失败，或不破坏领取证明系统和哈希函数的安全性，就无法重定向领取。

2.4 设计目标

1. **G1（枚举抗性）：** 给定一个目标标识符 e_B，$\mathcal{O}$ 无法确定哪些链上意图属于该标识符。 2. **G2（申领前不可关联性）：** 给定为同一标识符生成的两个领取前意图，在根据匹配的公开元数据进行条件分析后，$\mathcal{O}$ 无法仅凭链上数据确定它们是否具有同一接收方。 3. **G3（报价接受后的领取授权）：** 发送方接受有效报价后，如果某个意图存储了由此产生的元组（ρ_i、a_i、v_i、e_i），则只有能够生成相应 ZK-ACE 风格领取证明的一方才能领取该意图。 4. **G4（可移植恢复）：** 借助 Va-Dar，同一接收方身份可以跨设备重新派生，无需公开标识符目录。

5. **G5（加密隐匿性）：** 在集成式客户端流程中， A 和 B 均无需手动管理私钥、Gas 代币或特定区块链地址。

6. **G6（跨链）：** 支付可以在区块链 c_1 上发起，并在区块链 c_2上结算。

本文对 G1 和 G2 进行了形式化描述和分析。G3 由 HFIPay 的意图绑定逻辑与 ZK-ACE [2] 的授权保证组合而成；因此，我们明确说明这种组合，但不在此从第一原理重新证明 ZK-ACE 的安全性。因此，最好将这些隐私命题视为观察者模型证明草图，而非完整的基于博弈的归约。全文中，令 λ 表示部署安全参数，令 κ 表示实例化的域分离哈希的输出长度。我们的具体实例采用 256 位的 `intentId`值，因此，对于所关注的已部署系统，实际上有 $\lambda = 256$ 。我们用 H 表示抽象的域分离哈希函数；在证明概要需要之处，我们依赖其抗碰撞性（并在注明时依赖其抗原像性）。我们 并不声称 HFIPay 会隐藏明确公开的元数据，例如区块链、资产类型、金额、粗粒度纪元标签或时间信息。命题 5.1 和 5.2 则以这些公开元数据为条件；部署层的流量整形措施，例如批处理、标准金额区间或延迟通知，是可选的匿名性增强，而非协议要求。

3 协议

3.1 概述

注册接收方协议分为五个阶段——接收方设置、报价生成与验证、资金注入、声明，以及可选的退款。为清晰起见，图1省略了一次性设置阶段。令 d_{dep} 表示某个 HFI-Pay 部署（或跨虚拟机场景中的某个 n-VM 实例）独有的部署域标签。令 asset(c, a) 表示区块链 c上某项资产的规范链上编码。我们用 H 表示对所列字段的规范字节编码计算得出的抽象协议哈希。具体实现可以采用区块链原生的原语来实例化 H ，例如以太坊虚拟机上的 `keccak256` ，但 MUST 保持字段顺序和协议域分离标签完全一致。授权域标签使用 `hfipay:*` 命名空间；实现本地的 PDA 或存储种子可以使用不同的字节串，因为它们不是对重放敏感的授权消息。实现 MUST 确保链下客户端、链上合约和零知识电路对每条授权消息使用的字节编码完全相同；表1列出了规范字段顺序。

表1：授权消息的规范字段顺序。

消息	**规范字段顺序（从左到右）**
声明	tag, d_{dep}, c, a, e, intentId, ρ, v, β, T_{exp}, n
跨链领取	tag, d_{dep}, c_s, a, e, c_d, intentId, ρ, v, β_{c_d}, T_{exp}, n
退款	tag, d_{dep}, c, a, intentId, ρ, v, γ_A, T_{exp}

其中， c 表示采用部署特定编码的链标识符（例如，以太坊虚拟机上的 EIP-155`chainId` ，或 Solana 上的创世哈希前缀）， a 表示资产描述符， e表示粗粒度纪元标签。每个字段都序列化为定长或长度前缀

字节串；具体编码取决于部署，但必须由所有三个技术栈（客户端、合约、电路）达成一致。

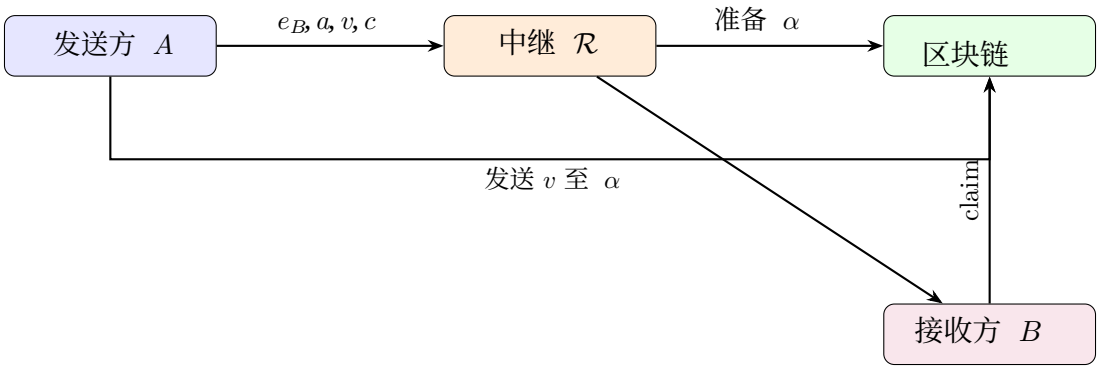


图 1：协议流程概览。

图 1 有意作了简化：中继至区块链的“准备 α”步骤涵盖了报价生成、任何资金注入前的意图注册，以及资金转移前发送方对已注册链上元组的验证。

3.2 接收方设置与绑定

此阶段针对每个接收方身份执行一次，并在绑定纪元变更时刷新。

1. 接收方 B 加载或重建 REV_B。在集成部署中，这可能来自本地密封工件，也可能通过 Va-Dar 在新设备上恢复。2. B 推导出身份承诺 $\mathrm{ID}_{\mathsf{com}\,B}$，并选择当前绑定纪元 e。3. B 推导出按纪元限定的绑定句柄 $u_{B,\,e}$ 以及相应的绑定密钥承诺 $K_{B,\,e}$。4. 在基线模式下，应用程序验证对 $\mathrm{Norm}(e_B)$ 的控制权，并插入相应的私有目录条目。5. 在验证报价模式下，B 还会获取一份证明 τ_B，其内容涵盖规范化标识符、绑定密钥承诺、绑定纪元和有效期限。只有当该证明有效且处于当前有效期内时，中继才会接受或刷新目录条目。6. 中继保留当前绑定纪元；如有必要，也会保留完成已注资意图相关索赔所需的任何尚未结算的先前纪元。

发送方无需知道 $\mathrm{ID}_{\mathsf{com}\,B}$ 或 $u_{B,\,e}$。在验证报价模式下，发送方只能看到 $(K_{B,\,e},\ \tau_B)$，以及一份报价证明，用于将报价的 ρ_i 与同一个隐藏句柄关联起来。

3.3 报价生成与意图创建

当发送方 A 发起支付，向电子邮件地址 e_B 支付资产 a，金额为 v，并在区块链 c上进行时，以下步骤描述了单个意图的处理过程。提及本地计算的值时，我们写作 ρ；将其列为索引元组（例如报价或观察者视图）的元素时，我们写作 ρ_i；对于正在构建的意图，两者表示相同的值。

1. A通过经过身份验证的 API 调用向中继 $\mathcal{R}$ 提交（e_B，a，v，c，γ_A，T_{exp}），其中 γ_A 是可选的退款目的地，T_{exp} 是过期时间。

2. $\mathcal{R}$ 查询 Norm(e_B) 对应的私有目录条目，获取当前绑定元组（ID$_{\mathsf{com}\,B}$，$u_{B,\,e}$，$K_{B,\,e}$，e）以及可选元数据，例如 τ_B。

3. $\mathcal{R}$ 生成一个密码学安全的随机意图标识符：

$$\texttt{intentId} \xleftarrow{\$} \{0,1\}^{256}.$$

4. $\mathcal{R}$ 计算确定性支付地址：

$$\alpha = \mathsf{DeriveAddress}(c, \texttt{intentId}).$$

5. $\mathcal{R}$ 计算特定于意图的盲化接收方绑定：

$$\rho \leftarrow H(\texttt{"hfipay:bind"} \| u_{B,e} \| \texttt{intentId}).$$

6. 在验证报价模式下，$\mathcal{R}$ 为一个关系Φ_{quote}构造链下报价证明 π_i^{quote}，该关系的公开输入包括（ρ_i，$K_{B,\,e}$，$\texttt{intentId}_i$），其见证证明

$$K_{B,e} = H(\texttt{"hfipay:bind-key"} \| u_{B,e}),$$
$$\rho_i = H(\texttt{"hfipay:bind"} \| u_{B,e} \| \texttt{intentId}_i).$$

中继或目录服务持有此见证，并为发送方生成 π_i^{quote}；接收方在报价创建期间无需在线。此处，τ_B 证明标识符与$K_{B,\,e}$ 之间的绑定，而 π_i^{quote} 证明报价 ρ_i 源自同一个隐藏句柄。正如威胁模型中所述，此验证报价模式仍依赖于绑定签发方以及生成 τ_B的注册流程的诚信。

7. $\mathcal{R}$ 返回一份报价

$$q_i = (\texttt{intentId}_i, \alpha_i, \rho_i, a_i, v_i, c_i, e_i, \gamma_{A,i}, T_{\mathsf{exp},i}, K_{B,e_i}, T_{\mathsf{bind},i}, \tau_B, \pi_i^{\mathsf{quote}}, T_{\mathsf{quote}}),$$

其中 T_{quote} 为报价过期时间。

8. 在验证报价模式下，A 会根据 (Norm(e_B)$K_{B,\,e_i}$，e_i，$T_{\mathsf{bind},\,i}$) 验证 τ_B，验证π_i^{quote}，检查报价时效性，在本地重新计算 $\alpha_i = \mathsf{DeriveAddress}(c_i,\ \texttt{intentId}_i)$，并确认报价字段（$a_i$，$v_i$，$c_i$，$\gamma_{A,\,i}$，$T_{\mathsf{exp},\,i}$）与发送方请求的支付相符。在基线模式下，$A$ 信任 $\mathcal{R}$，由其保证 ρ_i 的正确性。

9. 如果启用了退款， A 的客户端会对以下内容签署退款授权

$$\sigma_A^{\mathsf{ref}} \leftarrow \mathsf{Auth}_A\big(\texttt{"hfipay:refund"} \| d_{\mathsf{dep}} \| c \| a \| \texttt{intentId} \| \rho \| v \| \gamma_A \| T_{\mathsf{exp}}\big),$$

使用与 γ_A 关联的发送方控制的凭证或智能账户策略。若启用退款，则 $h^{\mathsf{ref}} = H(\sigma_A^{\mathsf{ref}})$；否则 $h^{\mathsf{ref}} = \perp$。完整的授权 σ_A^{ref}由发送方客户端保留，并可选择托管于 $\mathcal{R}$；只有其承诺 h^{ref} 存储在链上。退款时，提交方公开σ_A^{ref}，链上程序根据已存储的意图字段（c， a， intentId， ρ， v， γ_A， T_{exp}）重建预期的退款消息，验证 σ_A^{ref}是否符合 γ_A针对该消息的授权策略，并检查 $H(\sigma_A^{\mathsf{ref}}) = h^{\mathsf{ref}}$。如果发送方客户端和 $\mathcal{R}$ 都丢失了 σ_A^{ref}，退款路径将不可用；此后，该意图将一直处于 EXPIRED 状态，直至适用部署特定的恢复策略（如有）。

10. $\mathcal{R}$ 注册链上意图元数据（intentId， ρ， a， v， e， T_{exp}， γ_A， h^{ref}）或其特定于链的等效内容。

11. 在注册作为独立交易执行的部署中， A 在注资前必须确认， intentId 可读取的链上元组与（ρ_i、 a_i、 v_i、 e_i、 $T_{\mathsf{exp},\,i}$、 $\gamma_{A,\,i}$、 h^{ref}）完全一致。只要在转移资金的交易中绑定同一元组，等效的原子化创建并注资流程也可以接受。

12. $\mathcal{R}$ 存储私有记录（e_B、 intentId、 ρ、 a、 v、 c、 e、 γ_A、 T_{exp}、 t），并向 A返回最终资金注入指令。

关键设计要点是，在验证报价模式下，发送方会在注资前验证报价绑定和完整的报价意图元组，而区块链会在注资前或注资时提交（ρ、 a、 v、 e、 T_{exp}、 γ_A、 h^{ref}）。此后，中继无法为同一已注资意图替换接收方身份、换用其他报价资产，或更改到期与退款语义，否则就会导致发送方验证失败或改变链上状态。

上述“经过身份验证的 API 调用”应从操作层面理解，而不应狭义地从密码学角度理解。部署时应要求发送方身份验证或采用费用担保的报价创建，对报价签发进行速率限制，清理未注资意图，并将接收方通知推迟到确认资金已到账之后（或在支付不可退款的中继费用之后），以限制垃圾信息和通知滥用。

地址派生函数 DeriveAddress 特定于区块链：

定义 6 (EVM 地址派生)。 对于兼容 *EVM* 的链：

$$\alpha_{EVM} = \mathsf{CREATE2}(\mathit{factory}, \mathit{intentId}, \mathsf{keccak256}(\mathit{initcode}))$$

其中， $\mathit{factory}$ 是已部署的HFIPAY 工厂合约， $\mathit{initcode}$ 是 *EIP-1167* 最小代理创建代码。工厂合约还会存储与 ρ、资产描述符 a、金额 v、纪元标签 e、到期时间及退款信息对应的每个意图的元数据。

定义 7 (Solana 地址派生)。 对于 *Solana*：

$$\alpha_{SOL} = \mathsf{FindPDA}(\mathit{programId}, [\mathit{"hfipay"} \| \mathit{intentId}])$$

其中 FindPDA 是 *Solana* 标准的程序派生地址算法。相应的状态 *PDA* 存储 ρ、报价资产描述符 a、报价金额 v、纪元标签 e、到期时间、退款元数据，以及所需的任何金库引用。

我们用 $\mathrm{Auth}_A(m)$ 表示发送方提供的适用于链的退款授权。相比之下，领取授权由零知识证明提供，而非通过直接对接收方标识符进行签名来提供。

3.4 资金注入

接受报价后，发送方 A 会在区块链 c上，将 v 单位的报价资产 a 转账至地址 α 。

在以太坊虚拟机链上，即使账户合约尚未部署，只要工厂合约已记录意图元数据，α 就可以接收资金，因为 CREATE2 地址是确定性的。在 Solana 上，中继 $\mathcal{R}$ 会在 A 转账之前预先创建状态 PDA（并支付租金）。对于 SPL 资产， $\mathcal{R}$ 还会为每个受支持的铸币账户 M预先创建一个 PDA 所有的代币金库$\mathcal{V}_M$ 。

算法 1 资金注入阶段

要求： 发送方 A、地址 α、盲化绑定 ρ、报价资产 a、报价金额 v，区块链 c，可选 SPL 铸币账户 M

如果 c 兼容 EVM **则**
  $\mathcal{R}$ 调用 `registerIntent(intentId, ρ, a, v, e, T_exp, γ_A, h^ref)`
  A 调用 `transfer(`v`)` 以 α ▷ ETH 或 ERC-20
否则如果 c 是 Solana 且资产是 SOL **则**
  $\mathcal{R}$ 调用 `create_state(intentId, ρ, a, v, e, T_exp, γ_A, h^ref)`
  A 向 α转账 v Lamport
否则，如果 c 是 Solana，且 asset 是 SPL 代币 M **那么**
  $\mathcal{R}$ 调用 `create_state(intentId,` $\rho, a, v, e, T_{\mathsf{exp}}, \gamma_A, h^{\mathsf{ref}}$`)`
  $\mathcal{R}$ 调用 `create_vault(intentId,` M`)` ▷ 创建 PDA 所有的代币账户 $\mathcal{V}_M$
  A 向 $\mathcal{V}$转账 M 的 v 个单位 $_M$
结束条件

3.5 领取与退款

当接收方 B 得知存在待处理意图（例如通过电子邮件通知或中继查询）时，申领流程如下:

授权接口概要。 HFIPay 仅通过一个精简接口使用 ZK-ACE。在领取时，验证者会收到申领关系中已列出的公开元组，以及证明 π_i。索款方的见证包含确定性身份根（或可据此重新推导出该身份根的材料）、$\mathsf{ID}_{\mathsf{com}\,B}$ 对应的开启信息，以及满足盲化绑定关系所需的纪元范围句柄。Va-Dar 与此无关：它仅用于在不同设备间恢复相同的确定性身份根，而不用于生成报价证明。

1. B 加载或恢复 REV_B，从意图元数据（或保留的报价记录或通知）中读取纪元标签 e_i ，并重新推导 $\mathsf{ID}_{\mathsf{com}\,B}$ 以及 $u_{B,\ e_i}$。 2. B 选择一个目标地址 β_i （或者由集成客户端在本地推导一个新的地址）。

3. B 构成一条认领消息

$$m_i \leftarrow H(\texttt{"hfipay:claim"} \| d_{\mathsf{dep}} \| c_i \| a_i \| e_i \| \texttt{intentId}_i \| \rho_i \| v_i \| \beta_i \| T_{\mathsf{exp}} \| n_i),$$

其中 n_i 是防重放随机数。

4. B 为关系 Φ_{claim} 生成 ZK-ACE 申领证明 π_i，其公开输入包括 ($\mathsf{ID}_{\mathsf{com}B}$, ρ_i, a_i, e_i, $\texttt{intentId}_i$, v_i, β_i, T_{exp}, n_i)，其见证证明：

(a) $\mathsf{ID}_{\mathsf{com}B}$ 是由 B持有的确定性身份根对应的有效已注册承诺；(b) $u_{B,\ e_i} = H(\mathrm{Derive}(\mathrm{REV}_B,\ \mathrm{Ctx}_{\mathsf{bind}} \| e_i))$；(c) $\rho_i = H(\texttt{"hfipay:bind"} \| u_{B,\ e_i} \| \texttt{intentId}_i)$；以及 (d) $\mathsf{ID}_{\mathsf{com}B}$ 背后的身份根据 ZK-ACE 的防重放规则授权了 m_i。

5. 任何人（通常是 $\mathcal{R}$）都可以提交申领交易

$$\texttt{claim}(\texttt{intentId}_i, \mathsf{ID}_{\mathsf{com}B}, a_i, v_i, \beta_i, T_{\mathsf{exp}}, n_i, \pi_i).$$

链上程序会检查意图对应的已存储盲化绑定、已存储资产描述符、已存储纪元标签和报价金额是否等于 (ρ_i, a_i, e_i, v_i)，验证证明，然后要么将恰好 v_i 单位的 a_i 直接释放给 β_i，要么立即执行后续的中继提现，将恰好 v_i 单位的 a_i 提现至 β_i。

6. 释放交易在链上成功并达到所需的确认深度后，$\mathcal{R}$ 会在其链下数据库中将该意图标记为已申领。

由于证明将 $\texttt{intentId}_i$、ρ_i、a_i、v_i和 β_i绑定在一起，该协议具有抗抢先交易能力：矿工或内存池观察者可以复制该交易，但验证成功后，报价资产和金额仍只会发送至 β_i，而不会发送至攻击者指定的地址。

如果存款地址意外收到超出报价金额 v_i的资金，这部分盈余不属于预期的申领关系，需要单独的恢复策略。申领路径本身不会悄然将额外余额一并转给索款方。

退款仍需发送方授权。如果某个意图到期时尚未被申领，任何人都可以使用 σ_A^{ref} 提交预授权退款交易，链上程序会将资产 a_i 中恰好为报价金额 v_i 的部分释放给 γ_A。意外发送到存款地址、超出 v_i 的任何盈余均不在退款授权范围内，需要单独的恢复策略，这与上文所述的申领侧处理一致。

引理 3.1（从报价到申领的组合）。 在经验证报价部署中，假设发送方在验证了（$K_{B,\ e_i}, T_{\mathsf{bind},\ i}, \tau_B, \pi_i^{\mathsf{quote}}$）后接受报价，其中（$\mathit{intentId}_i, \rho_i$）固定；之后，区块链针对相同的（$\mathit{intentId}_i, \rho_i, e_i$）接受了声明证明 π_i。在报价证明系统和声明证明系统均具可靠性，且域分离哈希 H 对形式为（$\mathit{tag} \| u \| \mathit{intentId}$）的结构化输入具有抗碰撞性的条件下，两次成功验证所依据的必定是同一个纪元范围内的隐藏句柄 $u_{B,\ e_i}$。因此，成功申领的身份恰好属于其绑定密钥承诺已在已接受报价中得到证明的确定性身份族，尽管$\mathsf{ID}_{\mathsf{com}B}$本身在报价时无需向发送方披露。

证明概要。 报价证明确立了存在某个 u ，使得

$$K_{B,e_i} = H(\texttt{"hfipay:bind-key"} \| u) \quad \text{and} \quad \rho_i = H(\texttt{"hfipay:bind"} \| u \| \texttt{intentId}_i).$$

声明证明确立了某个确定性身份根 REV_B 的存在，其纪元范围内的派生值

$$u' = H(\mathsf{Derive}(\mathsf{REV}_B, \mathsf{Ctx}_{\mathsf{bind}} \| e_i))$$

满足相同的关系

$$\rho_i = H(\texttt{"hfipay:bind"} \| u' \| \texttt{intentId}_i).$$

由于 H 在结构化输入族 $(\texttt{"hfipay:bind"} \| \cdot \| \texttt{intentId}_i)$ 上的抗碰撞性（其中域标签和 $\texttt{intentId}_i$ 固定），要找到两个不同的句柄 $u \neq u'$，使其产生相同的 ρ_i ，就必须在 H 的可变部分 u上找到碰撞；根据生日界，成功概率至多为 $O(2^{-\kappa/2})$。因此， $u = u'$，且该声明绑定到发送方经验证的报价中出现的同一经证明的隐藏句柄。

□

算法 2 声明阶段（以太坊虚拟机）

要求： 接收方 B，意图 $(\texttt{intentId}, \rho, a, v, e, c)$，目标地址 β

1: $\alpha \leftarrow \mathrm{CREATE2}(\texttt{factory}, \texttt{intentId}, \mathrm{keccak256}(\texttt{initcode}))$
2: $m \leftarrow \mathrm{keccak256}(\texttt{"hfipay:claim"} \| d_{\mathsf{dep}} \| c \| a \| e \| \texttt{intentId} \| \rho \| v \| \beta \| T_{\mathsf{exp}} \| n)$ ▷ c 编码 chainId
3: B 重新推导 $\mathsf{ID}_{\mathsf{com}B}$ 和 $u_{B,e}$
4: $\pi \leftarrow \text{ZK-ACE}.\mathtt{Prove}(\mathsf{ID}_{\mathsf{com}B}, \rho, a, e, \texttt{intentId}, v, \beta, T_{\mathsf{exp}}, n)$
5: $\mathcal{R}$ 调用 $\texttt{factory.claimWithProof}(\texttt{intentId}, \mathsf{ID}_{\mathsf{com}B}, a, v, \beta, T_{\mathsf{exp}}, n, \pi)$
6: 工厂合约检查已存储的 (ρ, a, e, v)，验证 π，并向 β准确释放 v 单位的 a

在 Solana 上，会针对状态 PDA 验证相同的逻辑关系：

算法 3 声明阶段（Solana）

要求： 接收方 B，包含（$\texttt{intentId}, \rho, a, e, v$）的意图状态 PDA

1: B 重新推导 $\mathsf{ID}_{\mathsf{com}B}$ 和 $u_{B,e}$
2: $\pi \leftarrow \text{ZK-ACE}.\mathtt{Prove}(\mathsf{ID}_{\mathsf{com}B}, \rho, a, e, \texttt{intentId}, v, \beta, T_{\mathsf{exp}}, n)$
3: $\mathcal{R}$ 调用 $\texttt{claim}(\texttt{intentId}, \mathsf{ID}_{\mathsf{com}B}, a, v, \beta, T_{\mathsf{exp}}, n, \pi)$
4: 程序验证存储的（ρ, a, e, v），并准确转账 v 个报价中的 SOL 单位，或 SPL 资产到 β

3.6 意图生命周期

每个 intentId 均为一次性使用，并会沿着两种终止路径之一推进：

$$\texttt{CREATED} \to \texttt{FUNDED} \to \texttt{CLAIMED}, \qquad \texttt{FUNDED} \to \texttt{EXPIRED} \to \texttt{REFUNDED}$$

只有在 T_{exp} 之后仍无人认领时，意图才会进入 EXPIRED 。如果在创建意图时提供了退款目的地 γ_A ，则任何人都可以在到期后使用已存储的授权σ_A^{ref}提交预授权退款交易，链上程序会将报价资产的准确报价金额 v 释放至 γ_A。这样便完成了生命周期，同时不会让中继单方面控制已过期资金。

4 通过N-VM实现跨链支付

4.1 N-VM架构

n-Vm 链 [8] 是一条第一层区块链，在统一身份层下承载 n 个共存的虚拟机。单个身份承诺 `id_com` 可确定性地为每个受支持的虚拟机派生地址：

$$\alpha_{\mathrm{EVM}} = \mathsf{SHA\text{-}256}(\texttt{"evm:"} \| \texttt{id_com})[12{:}32] \tag{1}$$
$$\alpha_{\mathrm{SVM}} = \mathsf{SHA\text{-}256}(\texttt{"svm:"} \| \texttt{id_com}) \tag{2}$$
$$\alpha_{\mathrm{BVM}} = \mathsf{SHA\text{-}256}(\texttt{"bvm:"} \| \texttt{id_com}) \tag{3}$$
$$\alpha_{\mathrm{native}} = \texttt{id_com} \tag{4}$$

n-Vm 运行时包含一个统一代币账本，它在同一底层余额存储之上提供兼容 SPL 和 ERC-20 的接口。完整的 n-Vm 架构——包括基于操作码的分派、跨 VM 身份派生、统一代币语义和并行执行——详见配套论文 [8]。

4.2 跨链电子邮件支付流程

当源链 c_s 与目标链 c_d 不同时，HFIPay 会通过 n-Vm 层进行路由。我们用 X 表示具体的源链资产（对应基础协议中使用的抽象描述符 a）：

1. **存入**。发送方 A 将资产 X 存入源链 c_s 上的 n-Vm 桥接地址。

2. **包装**。n-Vm 代币运行时铸造 `wX`（包装后的 X），并将其锁定在与（`intentId`、ρ、c_s、a、e、v）关联的意图状态中：

$$\mathsf{mint}(\texttt{wX}, \alpha_{\texttt{intentId}}, v)$$

3. **领取**。接收方 B 选择目标虚拟机 c_d 和地址 β_{c_d}，然后构造一条跨虚拟机领取消息

$$m_d \leftarrow H(\texttt{"hfipay:cross-claim"} \| d_{\mathsf{dep}} \| c_s \| a \| e \| c_d \| \texttt{intentId} \| \rho \| v \| \beta_{c_d} \| T_{\mathsf{exp}} \| n_d)$$

以及一份 ZK-ACE 证明，证明生成 ρ 的私有句柄所属的同一确定性身份授权了 m_d。随后，任何一方都可以提交

$$\texttt{claim}(\texttt{intentId}, \mathsf{ID}_{\mathsf{com}\,B}, c_d, v, \beta_{c_d}, T_{\mathsf{exp}}, n_d, \pi_d)$$

并且，n-Vm 运行时仅在根据存储的元组（ρ、c_s、a、e、v）验证 π_d 后，才将释放权限绑定到 β_{c_d}。目标链资产形式（例如原生 X 或包装后的 `wX`）由运行时根据存储的源资产描述符 a通过解包逻辑确定；它不是声明证明的独立公开输入。

4. **解包**。代币运行时会销毁 `wX`，并将原生 X 释放到 c_d 上绑定的目标地址：

$$\mathsf{burn}(\texttt{wX}, v) \;\rightarrow\; \mathsf{release}(X, \beta_{c_d}, v)$$

如果包装后的意图过期且未被声明，则适用基础协议中使用的相同预授权退款策略，退款至源端退款目的地 γ_A。

本节假设 n-VM 部署具有共享状态和单一部署域标签 d_{dep}。如果在彼此独立的独立部署之间实现类似的路由，则必须在认领消息中准确包含部署域标签和源链标识符，以防止在不同部署之间发生重放；否则，这些部署可能会重复使用 `intentId`。

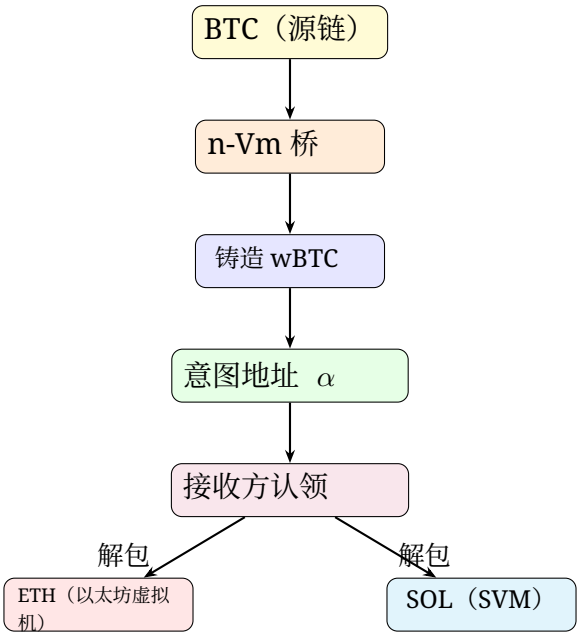


图 2：通过 n-Vm 实现的跨链支付路由。

这种机制无需在 n-Vm 系统本身之外再设立单独的 外部桥信任域：申领关系、封装/解封装以及目的地路由都在同一个共识边界内执行。不过，外部链存入的验证仍然是一项独立的信任假设：n-Vm 系统仍须验证发送方在源链上的存入确实已经发生（例如，通过轻客户端证明、验证者证明或等效机制）。因此，这种架构优势并非消除桥信任，而是将其整合：桥验证被纳入 n-Vm 共识内部，而非委托给额外的外部多重签名委员会；无论支付发生在同一条链上还是跨链，HFIPay 认领关系本身都保持不变。

4.3 比特币集成

比特币是持有最广泛的加密货币，但其 UTXO 模型以及缺乏可编程账户的特点，使标识符路由声明尤其难以实现。在 n-Vm BVM 模块下，同样可以采用 HFIPay 模式：

1. A 向 n-Vm BVM 存款地址发送 BTC。2. n-Vm 代币运行时铸造 wBTC，并将其锁定，以对应包含 ρ 的意图状态。3. B 使用相同的盲化绑定关系进行声明，并选择输出链。

4. 如果 B 选择 BTC：销毁 wBTC，通过 BVM 释放原生 BTC。

5。如果 B 选择 EVM 或 SVM：销毁 wBTC，并在目标 V 上释放为 wBTC M.

因此，接收方无需直接管理 UTXO 集即可接收比特币。具体的 BVM 实现方式——无论是基于比特币侧链、乐观汇总，还是对比特币脚本进行零知识验证——都与 HFIPay 协议本身无关，因此留待未来工作处理。

5 隐私分析

5.1 链上数据暴露

我们列举观察者 $\mathcal{O}$ 仅凭链上数据能够了解到的信息。

表 2：链上数据的可见性。

Data	链上?	会暴露身份吗?
`intentId`	Yes	不会（随机值）
盲化绑定 ρ_i	Yes	不会（一次性盲化值）
支付地址 α	Yes	否（源自随机意图）
资产标识符 a_i	Yes	仅披露资产类型
金额 v	Yes	N/A
纪元标签 e_i	Yes	否，如果取自粗粒度公共时间安排，而非收件人专属随机数
领取目的地 β （领取后）	Yes	仅地址[1]
身份承诺 $\mathsf{ID}_{\mathsf{com}B}$ （领取时，如果由验证器接口)	有时	仅承诺
电子邮件 e_B	**No**	—
$e_B \to (\mathsf{ID}_{\mathsf{com}B}, u_{B,e}, K_{B,e})$ 目录条目	**No**	—
时间戳	Yes	N/A

5.2 枚举抵抗性 (G1)

定义 8 (观察者视图)。 设 $\mathcal{O}$ 为一名观察者，可读取所有链上状态，并知晓目标电子邮件 e_B。 $\mathcal{O}$ 的视图为

$$\mathsf{View}_{\mathcal{O}} = \{(\mathit{intentId}_i, \alpha_i, \rho_i, a_i, v_i, e_i, t_i, \mathit{logs}_i)\}_i,$$

其中每个元组都可在链上公开观测。

[1]将 β 或 $\mathsf{ID}_{\mathsf{com}B}$ 关联到真实身份，需要交易所客户身份识别数据、入驻记录或已披露的绑定证明等外部信息。

De定义 9 （G1 元数据条件挑战）。 固定一个公开元数据元组 e

$$m = (c, a, v, e, t, \texttt{status})$$

以及目标标识符 e_B。挑战者随机抽取一个比特 $b \$ - \{0,1\}$、一个新鲜的 $\texttt{intentId}^\star \$ \leftarrow \{0,1\}^\lambda$，并计算

$$\alpha^\star = \mathsf{DeriveAddress}(c, \texttt{intentId}^\star).$$

若 $b = 0$，则使用目标接收方的隐藏句柄 $u_{B,\ e}$；若 $b = 1$，则使用独立抽样得到的对照接收方句柄 $u_{B',\ e}$，并采用相同的公共元数据元组 m。无论哪种情况，均设置

$$\rho^\star = H(\texttt{"hfipay:bind"} \| u \| \texttt{intentId}^\star)$$

对于选定的隐藏句柄 u，将 $(e_B, \texttt{intentId}^\star, \alpha^\star, \rho^\star, m)$ 发送给对手 $\mathcal{O}$，并接收其猜测 b'。枚举优势为

$$\mathsf{Adv}^{\mathsf{enum}}_{\mathrm{HFIPAY},\mathcal{O}}(\lambda) = \left| \Pr[b' = b] - \frac{1}{2} \right|.$$

命题 5.1 (枚举抗性)。 假设：*(i)* 中继的私有目录未遭到破坏；*(ii)* 无法从公共数据中恢复任何当前或保留的纪元句柄；*(iii)* H 被实例化为输出长度为 κ 位的域分离哈希，且每个 $\texttt{intentId}$ 均从 $\{0,1\}^\lambda$中均匀抽样。那么，对于上述 *G1* 实验中的任意多项式时间观察者 $\mathcal{O}$，

$$\mathsf{Adv}^{\mathsf{enum}}_{\mathrm{HFIPAY},\mathcal{O}}(\lambda)$$

除固定公开元数据元组m已包含的泄漏外，其概率为$\min(\lambda, \kappa)$ 可忽略。

证明概要。公链仅会揭示随机意图标识符、由其派生的地址、公共资产与金额元组、粗粒度纪元标签以及盲化绑定 $\rho_i = H(\texttt{"hfipay:bind"} \| u_{B,\ e} \| \texttt{intentId}_i)$。知道 e_B 并不会泄露任何有效或留存的纪元句柄，因为从规范化标识符到 $(\mathrm{ID}_{\mathsf{com}\,B},\ u_{B,\ e},\ K_{B,\ e})$ 的映射仅对中继目录可见，且发送方可验证报价仍保留在链下。没有$u_{B,\ e}$，观察者就无法检验某个公开元组是否属于 B，除非攻破中继目录或恢复接收方的确定性身份根。因此，链上元组在计算上独立于 e_B；但公开元数据（如资产类型、金额、粗粒度纪元标签和时间信息）会另外带来泄漏。

□

与朴素方案相比，其中 $\alpha = \mathsf{DeriveAddress}(c,\ \mathrm{keccak256}(e_B))$：任何知道 e_B 的一方都可以直接计算出接收方地址。

5.3 不可链接性 (G2)

定义 10 (认领前视图)。 令 视图$^{\mathsf{pre}}_{\mathcal{O}}$ 表示仅限于已注资但未认领意图的观察者视图。等价地，视图$^{\mathsf{pre}}_{\mathcal{O}}$ 不包含任何认领时披露信息，例如目标地址、验证者披露的身份承诺或认领后事件。

定义 11 (G2 元数据条件挑战**)**。 从某个部署明确公开的元数据分布中，选定两个公开元数据元组 m_1和 m_2 。挑战者抽取一个比特 $b \$ - \{0,1\}$。如果 $b = 0$，则针对同一接收方标识符 e_B 生成两个已资助的声明前元组，其隐藏句柄分别与 m_1 和 m_2一致。如果 b =1，则为 e_B 生成第一个元组，为一个独立的接收方标识符 $e_{B'}$生成第二个元组，同时确保它们分别匹配相同的公开元数据元组 m_1 和 m_2。在这两种情况下，所有 *intentId* 值均从 $\{0,1\}^\lambda$ 中独立抽取，挑战者将生成的声明前元组交给对手 $\mathcal{O}$，由其输出猜测 b'。不可链接性优势为

$$\mathsf{Adv}^{\mathsf{unlink}}_{\mathrm{HFIPAY},\mathcal{O}}(\lambda) = \left|\Pr[b' = b] - \frac{1}{2}\right|.$$

命题 5.2 (声明前地址不可链接性**)**。 在与命题 *5.1* 相同的假设下，*G2* 实验中的每个多项式时间观察者 $\mathcal{O}$ 都满足

$$\mathsf{Adv}^{\mathsf{unlink}}_{\mathrm{HFIPAY},\mathcal{O}}(\lambda)$$

一旦将比较的执行限定为具有相同的显式公开元数据元组 (m_1, m_2)，其优势便可忽略不计，且至多为 $\min(\lambda, \kappa)$ 。

证明概要。 每个意图都会独立采样一个 `intentId`。公开地址 α_i 仅由该随机值导出，而盲化绑定 ρ_i 是对隐藏纪元句柄 $u_{B,e}$ 与新鲜的 `intentId` 组合后所得内容的哈希。对于不掌握 $u_{B,e}$ 的观察者而言，元组 (α_i, ρ_i) 的分布与独立生成的一次性元组相同，唯一的限制是匹配的公开元数据。因此，在声明之前，不存在可公开观察到的结构能够区分“同一底层标识符使用两次”和“两个不同标识符各使用一次”。

□

命题 5.3 (声明后泄漏界限**)**。 声明后，重复使用同一公开目标地址 β ，并重复公开同一公开承诺 $\mathsf{ID}_{\mathsf{com}B}$ ，可能会暴露区块链身份层的共同控制权。如果验证器接口公开了 $\mathsf{ID}_{\mathsf{com}B}$，那么所有使用该相同公开承诺的索赔都可立即关联起来，即使它们的声明前意图无法关联也是如此。因此，关注声明后隐私的部署 *SHOULD* 通过特定于证明系统的验证接口隐藏 $\mathsf{ID}_{\mathsf{com}B}$ ，或者按照与注册信息刷新相协调的计划，定期更新面向用户的公开承诺。

命题 5.4 （已验证报价模式下的跨发送方可链接性）。 在验证报价部署中，同一绑定纪元内收到针对同一接收方的报价的所有发送方，都会在各自的报价中看到相同的绑定密钥承诺 $K_{B,\ e}$ 。因此，即使尚未发起声明，串通发送方也能确定自己在该纪元内支付给了同一接收方。这种跨发送方可链接性是纪元范围绑定模型固有的特性，不会影响链上观察者隐私（*G1*、*G2*），因为 $K_{B,\ e}$ 从未在链上提交。缩短绑定纪元可以缩小串通窗口，但代价是需要更频繁地刷新证明。

5.4 中继器泄露

如果 $\mathcal{R}$ 私有数据库遭到攻破后，对手会得知目录映射的

$$\mathsf{Norm}(e_B) \mapsto (\mathsf{ID}_{\mathsf{com}B}, \{u_{B,e}, K_{B,e}, e\}, \mathtt{metadata})$$

以及标识符到意图的映射。这使对手能够重建电子邮件到意图的关系图；如果存在公开的申领事件，还可以将其与认领目标和身份承诺关联起来。更严重的是，对于每个保留的纪元句柄 $u_{B,\ e}$ 对手都可以重新计算

$$H(\texttt{"hfipay:bind"} \| u_{B,e} \| \texttt{intentId}_i)$$

针对所涵盖时间窗口内公开的 `intentId` 值，事后将所有与该纪元关联的意图去匿名化。因此，影响范围是“所有受影响纪元的留存历史记录”，而不仅仅是当前待处理的交易图。

在隐私元数据方面，这与现有支付基础设施类似：PayPal、Stripe 和传统银行都会维护私有的发送方—接收方映射，一旦遭到入侵，这些映射就会暴露交易图。此外，由于 $\mathcal{R}$ 负责发送通知和进行中继，HFIPay 还存在应用层可用性依赖。相比之下，即使成功入侵 $\mathcal{R}$ ，也无法因此单方面获得认领权限，因为意图一旦在链上提交了盲化绑定，相关索赔仍必须满足对应确定性身份的 ZK-ACE 关系。

缓解措施包括：

- 使用硬件支持的密钥对静态映射数据库进行加密。
- 轮换绑定纪元，并在有限的保留期限后删除已结算纪元的句柄。

- 在保留期限过后，清除已认领的意图记录和过期的报价记录。
- 支持用户可见的通知、声明和披露操作审计日志。
- 在验证报价模式下使用经证明的注册记录和发送方可验证报价，使目录填充和报价生成可供审计。
- 将中继服务或目录分布在多个独立运营方之间（未来工作）。

即使在这种最坏情况下，公链本身仍保持不变：除非任意第三方也获取了遭到泄露的私有目录，否则他们仍只能看到假名化意图数据。

6 分层信任模型

关于基于标识符的加密货币支付，一个反复出现的担忧是：隐私可能要以牺牲可恢复性或合法问责为代价。HFIPay 通过将相关职责划分到三个层中，并明确注册模式，解决了这一问题。

6.1 三层架构

表3：分层信任架构。

层	隐私属性	问责机制
协议（链上）	随机意图、盲化绑定、报价资产元组，以及 ZK-ACE 申领证明；没有标识符会在链上出现直到声明之前	验证器逻辑、退款规则、精确金额释放和公开状态转换
应用程序 / 目录（$\mathcal{R}$）	从……开始的私密映射规范化标识符至（$\mathrm{ID}_{\mathrm{com}}$，$u_e$，$K_e$）和私有报价/意向记录	法律程序 $\rightarrow \mathcal{R} \rightarrow$ 日志，报价记录和目录状态
身份 / 绑定 ($\mathcal{I}$)	电子邮件或 OIDC 提供商验证标识符控制；验证报价模式会添加针对绑定密钥的证明承诺	法律程序 $\rightarrow$ 提供方或证明方 $\rightarrow$ 注册记录

不同部署模式下的信任划分略有不同：

- **基线部署**。 该应用程序在一次性绑定 $\mathrm{Norm}(e_B) \mapsto (\mathrm{ID}_{\mathrm{com}\,B},\ u_{B,\ e})$ 以及资金注入前正确执行逐意图接收方绑定方面均值得信赖。

- **经验证报价部署**。 绑定层对 $(\mathrm{Norm}(e_B)^{K_{B,\ e}},\ e)$ 提供证明，且发送方在资金注入前验证报价证明，从而消除中继单方面替换接收方的能力。

6.2 调查路径

针对具体调查，调查路径如下：

1. 从链下来源获取目标标识符。2. 通过法律程序 $\rightarrow$ 向电子邮件/OIDC 提供方或证明方 $\rightarrow$ 获取注册身份凭证。3. 通过法律程序 $\rightarrow$ 向中继/目录 $\rightarrow$ 调取与该标识符相关的意图记录。4. 将意图记录 $\rightarrow$ 与链上意图状态和申领事件 $\rightarrow$ 关联起来，追踪资金流。

这比普通的化名式 DeFi 提供了更清晰的审计路径，同时仍将标识符到意图的映射保留在公链之外。

6.3 与混币协议的比较

表 4：与混币/隐私协议的比较。

属性	混币器	标准去中心化金融	HFIPAY
发送方知道接收方吗?	No	通过地址	通过电子邮件
身份锚点	None	None	电子邮件
法律程序路径	None	困难	可通过 提供方和中继 记录
链上隐私（公开观察者）	Full	None	认领前隐私性强; 认领后隐私性较弱 如果相同的公开 承诺或 目标地址被 重复使用

7 实现方案概述

我们概述了适用于类以太坊和类 Solana 环境的兼容实现。本节的目的并非声称存在唯一且确定的实现方案，而是说明该协议仅需标准的区块链功能：确定性地址或状态账户、用于存储意图元数据的空间，以及所选 ZK-ACE 证明系统的验证者。本版本尚未提供端到端原型基准测试、报价证明大小、验证延迟或中继 API 测量结果。因此，我们在此介绍实现兼容性和状态布局，并将定量评估留待未来工作。

7.1 以太坊虚拟机：工厂合约、意图注册表与确定性存款

兼容 EVM 的实现可以将职责拆分到工厂合约/注册表合约和每个意图的存款账户中。

HFIPayFactory。工厂合约既是确定性地址预言机，也是意图注册表。它存储一条以 `intentId` 为键的 `IntentMeta` 记录，其中至少包含盲化绑定 ρ、报价资产描述符 a、报价金额 v、纪元标签 e、到期时间、退款元数据和生命周期状态。它通过 CREATE2 计算存款地址，并可在执行声明或退款时部署 EIP-1167 风格的最小代理克隆。主要入口包括：

- `registerIntent(...)`：在资金注入前，记录盲化绑定、报价资产、报价金额、纪元标签、到期时间和退款数据。
- `computeAddress(intentId)`：在链下预测存款地址，使注册后能够立即注入资金。

- `claimWithProof(...)`：检查存储的盲化绑定、报价资产、纪元标签、报价金额和 ZK-ACE 证明，然后向目标地址准确释放报价所指定的资产和金额。

- `refund(...)`：到期时间之后，根据已存储的意图字段重构退款消息，验证公开的 $\sigma_{\mathrm{ref}A}$ 是否符合 γ_A的授权策略及已存储的承诺 h_{ref}，并向 γ_A准确释放报价资产和金额。

领取语义。 以太坊虚拟机工程设计中的关键点是，资金注入后，中继无法更改预定的索款方，因为 `rho` 和完整的报价资产元组都在存入之前记录。对于经验证报价部署，发送方已在线下检查：报价的 `rho` 与经证明的接收方绑定密钥相匹配，报价地址与根据 `intentId`进行的确定性派生结果相匹配，并且可读注册表元组与已接受报价相匹配。因此，区块链会根据预先承诺的盲化值和准确的资产元组验证索款方，而不是在领取时依据中继方提供的标识符进行验证。

Gas 成本取决于所选的验证者，以及实现是直接重新释放资金，还是通过临时部署账户重新释放资金。这些细节取决于证明系统和合约设计，因此我们将其视为实现特定，而非协议定义性内容。

7.2 Solana：基于 PDA 的账户

在 Solana 上，相同的逻辑可以通过程序派生地址来实现，以由程序拥有的有状态账户取代合约部署。

每个意图的状态都存储在以 `[b"hfipay", intent_id]` 为种子的 PDA 中，其中包含 ρ、报价资产描述符、报价金额、纪元标签、到期时间、退款数据和状态标志。对于 SPL 代币支付，程序还会为每个受支持的铸币账户 M 创建一个 PDA 所有的代币金库 $\mathcal{V}_M$ 。索款指令会根据已存储的 (ρ, a, e, v) 元组验证所提供的 ZK-ACE 证明，然后将准确的报价 SOL 或 SPL 资产金额转账至索款目标地址。如果目标验证者在 Solana 上以整体方式执行的成本过高，部署时可以采用配套验证器程序、递归聚合，或针对 Solana 执行环境选择的证明系统。

与以太坊虚拟机路径不同，领取时不会额外部署合约；状态 PDA 和任何金库都在资金注入阶段创建。

7.3 中继服务

中继／目录层是唯一的链下组件，也是唯一持有标识符到意图映射的一方。从架构上看，它是由私有数据库支持的计算层：

- **目录表**：将 Norm(e_B) 映射到当前及尚未结算的历史绑定纪元（ID$_{\mathrm{com}B}$，$u_{B,\,e}$，$K_{B,\,e}$，e，`mode`，`metadata`），并附带注册时间、证明引用和轮换状态等元数据。

- **意图表**：存储私有意图记录，其中包含 `intentId`、ρ、资产描述、金额、区块链、纪元标签、退款数据、到期时间和状态。

- **意图 API**：接受（e_B，a，v，c，γ_A，$T_{\text{到期时间}}$），以私密方式解析目录条目，计算盲化绑定，并在验证报价模式下返回完整报价元组，以及（$K_{B,\,e}$，$T_{\text{绑定}}$ τ_B，$\pi_{\text{报价}\,i}$）和确定性存款地址。
- **交易中继**：代表用户提交声明和退款交易，并支付燃料费或租金，使发送方和接收方都无需持有原生代币。
- **通知**：仅在检测到资金到账后，或在接收方接受不可退款的中继费用后，才告知接收方有待处理付款，从而限制通知垃圾信息。
- **滥用控制**：通过实施发送方身份验证、速率限制和报价过期机制，减少无资金意图垃圾信息。

关键在于，中继不持有 REV_B，不托管资金，并且在盲化绑定注册到链上后，无法单方面重定向声明。因此，其信任角色比托管方更有限，但比纯粹的传输中继更广：在基线模式下，用户需要信任中继正确完成一次性目录注册和资金存入前的接收方绑定；而在验证报价模式下，发送方可验证报价消除了后一项信任假设。

7.4 复杂度与开销对比

即使没有基准测试数据，相较于简单的“标识符 $\rightarrow$ 地址”映射，该协议增加的开销也很容易描述。经验证报价部署中，每个已注资意图都需要生成一次链下报价证明，并进行一次发送方验证；基线部署和经验证报价部署都要求接收方生成一次声明证明，并在领取时进行一次链上证明验证。中继 API 路径会增加一次目录查找和一个意图注册步骤，而发送方路径会增加本地地址推导；在经验证报价部署中，还需验证已注册意图元组。

表5：与更简单的标识符路由方法进行定性比较。

方案	额外链下 Work	链上工作	隐私／信任权衡
朴素直接映射	除此之外没有标识符查找	普通转账 only	无法防范任何知道该标识符的人
HFIPay 基线	中继目录查询；可选发送方退款授权；接收方领取证明生成	意图注册加上一份领取证明验证	保留认领前观察者隐私，但中继仍然需要信任预融资接收方绑定
HFIPay 经验证的报价	所有基准成本加上一份报价证明由……生成中继和一份报价由以下主体验证发送方	相同的链上路径与基线相同	消除单方面的中继接收方替换之前资金注入，同时保留标识符解析在链下进行

发送方可见报价材料的字节大小为 $O(|\tau_B| + |\pi^{\text{quote}}|)$，领取交易大小为 $O(|\pi_i|)$。端到端延迟主要取决于目录查找、报价证明生成，以及领取前的一轮链上注册／资金注入。这些数值取决于所选用的 ZK-ACE 和报价证明系统，因此本文将数值评估留待未来的制品论文，而不硬编码实现特定的测量结果。

8 相关工作

隐形地址（ERC-5564）。隐身地址协议 [4] 使用发送方的临时密钥与接收方的隐身元地址进行椭圆曲线迪菲–赫尔曼密钥交换，从而推导出一次性接收地址。这在协议层实现了强不可链接性，但也带来了 HFIPay 所规避的两项成本：（i）接收方必须持续扫描所有链上交易以检测支付；（ii）发送方必须事先知道接收方的隐身元地址，而这本身就需要一种发现机制。HFIPay 将发现过程转移到应用层中继，并使用盲化申领绑定，而非发送方侧的椭圆曲线迪菲–赫尔曼密钥交换，以协议层面的去中心化为代价，换取更简单的标识符路由和领取时证明。

以太坊名称服务与基于电子邮件的命名。以太坊名称服务 [5]将易读的名称映射到区块链地址，解决了十六进制字符串晦涩难懂这一用户体验问题。然而，这种映射完全公开：任何知道以太坊名称服务名称的人都可以查看其所有者的余额和交易历史。HFIPay解决了同样的易用性问题，同时将标识符到意图以及标识符到地址的映射保密。因此，这两个系统互为补充：以太坊名称服务提供公开的持久化命名，而HFIPay提供私密的中继媒介路由。

托管式“发送至邮箱”系统。 中心化交易所和支付应用通常会提供通过电子邮件或手机号码发送的功能，并通过托管资金、在运营方私有数据库中解析标识符来实现。这些系统在运作方式上与HFIPay相近，因为它们依赖中继式目录，但信任的归属有所不同：运营方最终决定路由和资金释放。HFIPay则在保持标识符映射私密的同时，将资金注入后的授权交由可在区块链上验证的证明处理，从而在验证报价模式下缩小中继的权限。

ZK-ACE。 ZK-ACE [2]将随交易携带的签名对象替换为与身份绑定的零知识授权声明，并以确定性身份承诺作为支撑。HFIPay 使用 ZK-ACE 并非用于发送方路由，而是用于接收方索赔授权：中继私下解析标识符、提供可由发送方验证的报价，并将盲化绑定 ρ_i 与报价资产元组一同承诺后，接收方即可在链上证明，同一已承诺身份授权将该特定意图发放至某个目标地址。通过这种组合，意图注册后，索赔正确性便不再由中继自行决定。

Va-Dar。 Va-Dar [3] 为确定性身份提供密钥化发现和口令保护的恢复，同时不暴露公开标识符注册表。在 HFIPay 中，它提供接收方私有恢复路径：接收方可通过标识符和口令，在不同设备上重新派生出相同的确定性身份，进而重新派生公开身份承诺以及支付层使用的私密领取绑定句柄。由于已注资意图也存储了粗粒度纪元标签 e_i，恢复时无需实时中继仅为获知领取时必须重新派生哪个绑定纪元。然而，Va-Dar 本身无法解决发送方对接收方身份的公开查询问题；这仍由中继的私有目录负责。

账户抽象（ERC-4337）。账户抽象 [6] 支持智能合约钱包具备灵活的身份验证、Gas 赞助和批量操作。HFIPay 与之互补：它解决发现与路由问题，而账户抽象则在已知索款方或目标地址后，负责执行与赞助。因此，如有需要，以太坊虚拟机上的实现可以通过兼容 ERC-4337 的智能账户接口提供 HFIPay 声明。

Tornado Cash 与隐私池。混币协议 [7] 通过将存款汇集到共同的匿名集中，并允许用户通过零知识证明取款，切断发送方与接收方的关联。这能提供强大的链上隐私保护，但需要将存款混入共享池，且不提供内置的便于用户使用的路由层。HFIPay 则采用逐意图随机路由和盲化申领绑定，因此资金不会混入同一资金池，同时链下标识符锚点仍可通过中继和注册层使用。

9 结论

HFIPay 最适合被理解为由四个层构成的组合：私有标识符路由、发送方可验证报价绑定、盲化意图承诺，以及链上零知识领取授权。中继会生成一个全新的随机 `intentId`，并推导出一个一次性

存款地址，且只将意图专属盲化绑定 ρ_i 以及报价资产元组提交到链上。在经验证报价部署中，发送方会先检查一份链下经证明的报价，该报价证明报价的 ρ_i 是从预期接收方的隐藏绑定句柄推导而来；发送方在本地重新计算报价中的存款地址，并在注入资金前确认已注册的链上元组，整个过程无需公开标识符注册表。之后，接收方通过 ZK-ACE 证明：支撑某个已承诺链上身份的同一确定性身份能够满足该盲化绑定，并授权将报价资产及金额发放至选定目的地。

这种架构明确划定了隐私边界。在认领前，区块链会公开随机意图标识符、存款地址和一次性盲化绑定，但不会公开人性化标识符或可重复使用的接收方标签。认领后，可关联性取决于公开身份承诺和目的地信息是否被重复使用。Va-Dar 通过为接收方提供一条私密的跨设备恢复路径，使其能够恢复至同一确定性身份，从而完善了整个方案，而无需公开标识符注册表。

由此形成的信任模型比托管方的信任模型范围更窄，也更加清晰。在基线部署中，应用程序仍需被信任，才能正确完成一次性注册和资金存入前的接收方绑定。在经验证报价部署中，经证明的链下报价消除了中继单方面替换资金存入前接收方的权力，同时保留了领取前的链上隐私。无论采用哪种部署，一旦意图的盲化绑定被提交到链上，中继就无法再单方面重定向申领。

与 n-Vm 运行时组合后，同一申领关系可扩展至跨链结算。这种扩展仍取决于底层运行时对外部存款的验证，但无需单独采用桥专属接收方授权模型。

9.1 未来工作

仍有几个方向有待探索：

- **报价证明优化**。 压缩或聚合可由发送方验证的报价证明，能够降低延迟，并降低经验证报价部署在大规模运行时的成本。

- **私有目录联邦**。 将私有目录分布到多个独立运营方，或通过联邦方式进行管理，可降低单点失陷风险，同时保留私有查询能力。

- **证明聚合**。 对 ZK-ACE 声明证明进行批量验证或递归验证，可降低高吞吐量支付枢纽的验证者成本。
- **延迟首次收款接入**。 为此前未注册的接收方制定完善的首次收款注册流程，仍是一个重要的部署问题。

参考文献

[1] J. S. Wang. ACE-GF：原子密码实体生成框架。arXiv 预印本 arXiv:2511.20505，2025。

[2] J. S. Wang。ZK-ACE：面向后量子区块链系统、以身份为中心的零知识授权。arXiv 预印本 arXiv:2603.07974，2026。

[3]J. S. Wang。VA-DAR：一种抗后量子密码学、供应商无关的确定性工件解析方案，用于无服务器、抗枚举的钱包恢复。arXiv 预印本 arXiv:2603.02690，2026。[4] T. Wahrstätter、M. Solomon、B. DiFrancesco 和 V. Buterin。ERC-5564：隐形地址。以太坊改进提案，2022。[5] N. Johnson。ERC-137：以太坊域名服务——规范。以太坊改进提案，2016。[6] V. Buterin、Y. Weiss、D. Tirosh、S. Nacson、A. Forshtat、K. Gazso 和 T. Hess。ERC-4337：使用替代内存池的账户抽象。以太坊改进提案，2021。[7] A. Pertsev、R. Semenov 和 R. Storm。Tornado Cash 隐私解决方案。白皮书，2019。[8] J. S. Wang。n-VM：一种具有共享身份和代币状态的多虚拟机第一层架构。arXiv 预印本 arXiv:2603.23670，2026。