Pagamentos Automatizados Programáveis em Carteiras Auto-Custodiadas: Uma Exploração Técnica

Ana Julia Bittencourt Fogaça[1], **Saulo Popov Zambiasi**[2]

[1]Universidade do Sul de Santa Catarina (UNISUL)
Tubarão - SC - Brasil

[2]Universidade do Sul de Santa Catarina (UNISUL)
Florianópolis - SC - Brasil

anajuliabit@gmail.com, saulopz@gmail.com

Abstract.

Resumo. *À medida que a tecnologia blockchain continua a ganhar adoção, percebe-se um crescimento na demanda por funcionalidades que viabilizem sua aplicação em situações reais. No âmbito deste artigo técnico, nos inspiramos na recente publicação da Visa como premissa para a exploração da implementação de pagamentos programáveis automatizados em carteiras de auto custódia. Adentramos na esfera do conceito de Abstração de Contas e propomos uma execução codificada em Solidity com o propósito de habilitar pagamentos recorrentes originados diretamente de carteiras sob controle do usuário. Por meio desta sondagem, nosso intuito primordial é fornecer perspectivas elucidativas e orientações pragmáticas relativas à implementação de pagamentos automáticos no contexto das finanças descentralizadas.*

1. Introdução

Primeiramente introduzida por Satoshi Nakamoto em 2008, a inovação da tecnologia blockchain tem sido reconhecida como uma das ”megatendências”computacionais com capacidade para reestruturar a paisagem global nas próximas décadas (World Economic Forum, 2015). A singularidade inerente à tecnologia blockchain, englobando aspectos de segurança, transparência e rastreabilidade, tem estimulado uma gama diversificada de setores industriais a explorarem sua utilização na reformulação de suas operações centrais. A amplitude das suas aplicações estende-se para além do domínio da criptomoeda, permeando domínios como a gestão de identidade, saúde, eleições governamentais, entre outros [Pu and Lam 2021].

Com a publicação do *whitepaper* do Ethereum em 2014[Buterin 2014], um marco significativo foi estabelecido no desenvolvimento da tecnologia blockchain. Em contraste com o Bitcoin, que foi inicialmente concebido como um sistema de pagamento digital, o Ethereum introduziu uma nova faceta à tecnologia blockchain: os Smart Contracts. Essa inovação foi projetada originalmente para se beneficiar dos atributos da blockchain, implementando automaticamente os termos acordados entre duas partes ao formalizar um contrato num ambiente desprovido de confiança, uma característica que conduziu à denominação de *Smart Contract* para esse código de software[Pinna et al. 2019].

A clara compreensão dos detalhes operacionais das contas no ecossistema Ethereum torna-se imperativa para um entendimento completo de como funcionam os Smart

Contracts. No Ethereum, existem essencialmente duas categorias distintas de contas: as *externally owned accounts* (EOAs) e as *contract accounts*.

As EOAs, ou contas de propriedade externa, podem ser estabelecidas através de carteiras como a MetaMask. Essas contas detêm uma chave privada que autoriza o controle sobre o acesso aos fundos ou contratos associados. Em contraste, as *contract accounts*, ou contas de contrato, apresentam características diferenciadas das EOAs. Uma conta de contrato abriga o código de um *smart contract*, uma funcionalidade ausente numa simples EOA. Além disso, uma conta de contrato não dispõe de uma chave privada. Ao invés disso, é detida (e controlada) pela lógica incorporada no código do seu *smart contract* - um programa de software gravado na blockchain Ethereum no momento de criação da conta de contrato e operado pela Ethereum Virtual Machine (EVM) [Antonopoulos and Wood 2018].

As contas de contrato possuem endereços, à semelhança das EOAs, e também são capazes de enviar e receber Ether. Contudo, quando uma transação é destinada a um endereço de contrato, ela provoca a execução desse contrato na EVM, usando a transação e os dados da transação como entrada. Além do Ether, as transações podem conter dados que especificam qual função particular do contrato deve ser ativada e quais parâmetros devem ser fornecidos a essa função. Assim, as transações podem invocar funções dentro dos contratos [Antonopoulos and Wood 2018].

Agora que elucidamos a distinção entre as contas de propriedade externa (EOAs) e as contas de contrato, podemos aprofundar nossa compreensão no contexto das aplicações descentralizadas (DApps). Essas aplicações substituem a infraestrutura tradicional de back-end por smart contracts operando em blockchains como o Ethereum. No entanto, a adoção desses DApps enfrenta o desafio premente de proporcionar uma experiência de usuário comparável àquela oferecida pelas aplicações centralizadas.

Para enfrentar esses desafios intrínsecos, surgem esforços incessantes visando aprimorar e simplificar a experiência do usuário no ambiente do Ethereum. Como o Ethereum constitui uma rede descentralizada e de código aberto, a comunidade formula continuamente propostas de melhoria conhecidas como *Ethereum Improvement Proposals* (EIPs). Uma proposta relevante para nossa discussão é a EIP-4337.[Buterin et al. 2021]

A EIP-4337, popularmente denominada como *Account Abstraction* (AA), propõe uma inovação radical no modelo convencional de contas do Ethereum, sugerindo a utilização de *smart contracts* no lugar das tradicionais EOAs. Essa proposta tem o potencial de desencadear a exploração de novos casos de uso na plataforma.

Na essência, a *Account Abstraction* busca consolidar contas de usuários e contratos inteligentes em um único tipo de conta Ethereum. Assim, as contas de usuários passariam a operar como contratos inteligentes. Este conceito, apesar de simples, traz implicações significativas: ele confronta e busca modificar os requisitos inflexíveis inerentes às transações Ethereum atuais, que são rigidamente codificados no protocolo Ethereum, tais como a necessidade de uma assinatura ECDSA válida, um nonce válido e saldo suficiente para cobrir os custos de computação [Andrew Beams and Kumaresan 2023].

Ao oferecer maior flexibilidade no processo de validação de transações, a Account Abstraction (AA) desencadeia uma série de novas capacidades. Entre alguns casos de uso, destacam-se a possibilidade de autorização personalizada, adaptando a lógica de

autorização de acordo com as necessidades e permitindo a escolha de esquemas criptográficos alternativos em caso de descoberta de vulnerabilidades no ECDSA, garantindo a segurança contínua (1). Além disso, a AA permite a abstração do custo do gás, possibilitando que transações iniciadas por contas de contrato sejam pagas pelo proprietário ou por um terceiro chamado "paymaster", que arca com as taxas de gás da transação (Buterin, 2021). Essa abordagem também facilita a entrada de usuários inexperientes, pois o paymaster pode ser financiado com moeda fiduciária fora da cadeia, eliminando a necessidade de lidar diretamente com ether. A AA ainda possibilita a implementação de carteiras com recuperação social, onde um contrato de conta pode ser programado com um endereço de backup controlado por uma parte confiável ou usando um esquema de múltiplas assinaturas, permitindo a recuperação da carteira em casos de perda de chaves privadas ou frases de backup. O caso de uso mais importante para o contexto deste artigo é a capacidade de realizar pagamentos automáticos, adicionando regras de validação que não dependem necessariamente de assinaturas (Visa, 2023). Isso é inédito para blockchains, uma vez que pagamentos automáticos são impossíveis em EOAs, pois assinaturas são necessárias para todas as transações.

Diante desse cenário emergente, a utilização da blockchain na esfera dos pagamentos programáveis automatizados desponta como foco central desta pesquisa. A Visa, reconhecida por suas soluções de pagamento, está empenhada em explorar métodos que simplifiquem o processo, permitindo aos usuários realizar auto-pagamentos na blockchain sem a necessidade de divulgar suas chaves privadas.

Neste trabalho, iniciaremos com a análise meticulosa da publicação recente da Visa sobre o assunto. Em seguida, aprofundaremos nosso entendimento do conceito de Abstração de Contas e, finalmente, sugeriremos uma implementação em Solidity que habilite a execução de pagamentos automáticos programáveis em carteiras auto-custodiadas. Com a implementação proposta, exploraremos as implicações e oportunidades que se apresentam para o emergente campo de pagamentos.

2. Revisão Bibliográfica

2.1. A publicação da Visa

Em sua recente contribuição para a expansão do campo de pagamentos automáticos programáveis em carteiras auto-custodiadas via blockchain, a Visa apresentou revelações significativas numa publicação técnica intitulada "Autopayments via Account Abstraction". As próximas linhas fornecem uma síntese das conclusões primárias e propostas emergentes desta publicação.

Na sua abordagem, a Visa introduz um mecanismo inovador que simplifica a possibilidade do usuário de executar autopagamentos, prescindindo do uso da chave privada associada à sua identidade. O intuito subjacente é viabilizar autopagamentos para comerciantes sem a necessidade de expor a chave privada do usuário a qualquer servidor de terceiros. Como alternativa, um contrato inteligente é capacitado para processar um autopagamento em nome do usuário para o comerciante destinatário, sem a exigência da chave privada do usuário.

O modelo concebido pela Visa concede ao contrato inteligente a autoridade de conduzir pagamentos automáticos aos comerciantes associados ao usuário, estando sempre condicionado à ratificação deste último. Esta ratificação pode ser obtida através do

fornecimento de dados que o contrato inteligente está autorizado a utilizar para realizar os autopagamentos em representação do usuário. Em essência, o usuário pode pré-autorizar a transação, habilitando o contrato inteligente a processar o pagamento em seu nome quando uma solicitação é encaminhada pelo comerciante. A Visa sugere ainda que o usuário possa compilar uma lista de permissões, onde poderá pré-autorizar transações com pagadores predeterminados, como comerciantes, outros usuários, entre outros.

Na monografia técnica mencionada, a Visa recorreu a um conceito recente e a uma das principais propostas de desenvolvedores do Ethereum conhecida como Abstração de Conta para investigar a implementação de contratos inteligentes que viabilizem pagamentos programáveis automáticos. Propuseram uma inovadora solução para uma aplicação real de pagamentos automáticos, demonstrando como estruturar um contrato inteligente para uma carteira autogerida capaz de retirar fundos automaticamente, sem necessitar da participação ativa do usuário em cada instante para instruir e realizar pagamentos numa blockchain.

Na seção Desenvolvimento deste artigo, exploraremos a proposta da Visa em maior detalhe, bem como as implicações e oportunidades que ela apresenta para o campo emergente de pagamentos. Iremos propor uma implementação escrita em Solidity, a linguagem de programação mais utilizada na plataforma Ethereum, que permitirá a execução de pagamentos automáticos programáveis em carteiras auto-custodiadas. Apesar da visa ter descrito a arquitetura de uma conta de carteira autogerida capaz de realizar pagamentos automáticos, não foi fornecida uma implementação de referência. A implementação proposta neste artigo visa preencher essa lacuna. Mas antes de entrarmos em detalhes sobre a implementação, precisamos entender o que é a Abstração de Contas É disso que trataremos na próxima seção.

2.2. Abstração de Conta

A Abstração de Conta é um conceito que vem sendo explorado para aumentar a flexibilidade e funcionalidade das contas na rede Ethereum.

Conforme destacado por Andreas M. Antonopoulos em ”Mastering Ethereum”[Antonopoulos and Wood 2018], duas categorias primárias de contas são estabelecidas na rede Ethereum: *Externally Owned Accounts* (EOAs) e *Contract Accounts*.

As EOAs, as contas de propriedade externa, podem ser instituídas através de interfaces de carteira como a MetaMask. Estas contas estão vinculadas a uma chave privada que proporciona controle sobre o acesso a fundos e contratos correspondentes. Em contrapartida, as *Contract Accounts*, ou contas de contrato, diferem fundamentalmente das EOAs. Uma conta de contrato hospeda o código de um *smart contract*, recurso ausente em uma EOA convencional. Além disso, uma conta de contrato não possui uma chave privada. Em vez disso, a posse (e controle) está estruturada em torno da lógica do código de seu *smart contract*, um programa de software que é registrado na blockchain Ethereum durante a criação da conta de contrato e operado pela Ethereum Virtual Machine (EVM) [Antonopoulos and Wood 2018].

As contas de contrato, assim como as EOAs, possuem endereços e têm a capacidade de enviar e receber Ether. No entanto, quando uma transação é direcionada a um endereço de contrato, isso aciona a execução desse contrato na EVM, usando a transação

e os dados da transação como entrada. Além do Ether, as transações podem incorporar dados que indicam qual função específica do contrato deve ser ativada e quais parâmetros devem ser fornecidos para essa função. Desta forma, as transações têm a capacidade de invocar funções dentro dos contratos [Antonopoulos and Wood 2018].

A Abstração de Conta sugere um desenvolvimento transformador na maneira como essas contas funcionam, propondo que todas as contas na rede Ethereum tenham a potencialidade de operar como Contas de Contrato. Isso implica que cada conta poderia abrigar sua própria lógica de operação, conduzindo a um grau de personalização e funcionalidade sem precedentes. Por exemplo, uma conta poderia ser programada para gerenciar transações de uma maneira específica ou para se defender contra certos tipos de ataques.

3. Desenvolvimento

3.1. O problema

Ilustração detalhada de um cenário hipotético que demonstra a necessidade de pagamentos automatizados em uma carteira auto-custodiada. Análise das limitações enfrentadas pelos usuários ao tentar agendar pagamentos automáticos em uma blockchain como o Ethereum.

3.2. Abordagem da solução

Visão geral da solução proposta, aproveitando a Abstração de Contas e contratos inteligentes. Introdução de contas delegáveis e do conceito de regras de validade programáveis para transações. Demonstração de como os pagamentos automáticos podem ser implementados usando contratos inteligentes de pagamento automático pré-aprovados.

3.3. Implementação técnica

Explicação dos passos técnicos necessários para configurar e configurar contas delegáveis em carteiras auto-custodiadas. Visão sobre o fluxo de transação e interação entre o contrato inteligente de pagamento automático e as carteiras controladas pelo usuário.

3.4. Vantagens e Potenciais Aplicações

: Discussão dos benefícios e vantagens oferecidos pela solução proposta para pagamentos automáticos. Exploração de casos de uso potenciais além de pagamentos recorrentes, como serviços de recuperação de conta de terceiros e gestão de ativos.

4. Conclusão

Reflexão sobre a importância dos pagamentos programáveis e o potencial para inovações futuras no espaço blockchain. Pesquisa Futura e Considerações Identificação de possíveis áreas para futuras pesquisas e desenvolvimento na automatização de pagamentos em carteiras de autoguarda. Discussão sobre possíveis desafios e considerações a serem tratados na implementação de pagamentos automáticos em escala.

Referências

Andrew Beams, Catherine Gu, S. R. M. M. and Kumaresan, R. (2023). Auto payments for self-custodial wallets. *Ethereum Project Yellow Paper*.

Antonopoulos, A. and Wood, G. (2018). *Mastering Ethereum: Building Smart Contracts and DApps*. O'Reilly Media.

Buterin, V. (2014). Ethereum white paper.

Buterin, V., Weiss, Y., Gazso, K., Patel, N., Tirosh, D., Nacson, S., and Hess, T. (2021). Erc-4337: Account abstraction using alt mempool [draft]. Ethereum Improvement Proposals.

Pinna, A., Ibba, S., Baralla, G., Tonelli, R., and Marchesi, M. (2019). A massive analysis of ethereum smart contracts empirical study and code metrics. *IEEE Access*, 7:78194–78213.

Pu, S. and Lam, J. S. L. (2021). Blockchain adoptions in the maritime industry: a conceptual framework. *Maritime Policy & Management*, 48(6):777–794.